Public Works Canada
Travaux publics Canada

Solar Programs Office
Bureau des programmes solaires

Public Works Canada
Solar Programs Office

SOLAR COMMISSIONING PACKAGE:
OPERATIONS MANUAL

June 1984

Prepared by:

Sheaff and Associates
Engineering Inc.
P.O. Box 1048
Manotick, Ontario
K0A 2N0

(613) 692-3506

Prepared for:

Solar Programs Office
Public Works Canada
Sir Charles Tupper Building
Riverside Drive
Ottawa, Ontario
K1A 0M2

(613) 998-9817

This is not a departmental document. Do not cite as reference or catalogue in library.

SOLAR COMMISSIONING PACKAGE OPERATIONS MANUAL

TABLE OF CONTENTS

Description	Page
1.0 INTRODUCTION	1-1
2.0 SCP SYSTEM HARDWARE SET-UP	2-1
2.1 System Components	2-1
2.2 Cable Connections	2-2
2.3 Power Concerns	2-2
2.4 Basic Hardware Functional Checks	2-3
2.5 Environmental Concerns	2-4
2.6 Calibration	2-5
3.0 SCP SOFTWARE	3-1
3.1 Overview	3-1
3.2 IBM DOS And DOS Command Summary	3-2
3.3 Disk File Description	3-6
3.4 Set-Up Vs. Run-Time Operation	3-12
3.4.1 The Run-time Menu	3-12
4.0 CONFIGURING THE MONITORING SYSTEM	4-1
4.1 Overview And System Generic Capabilities	4-2
4.2 Virtual Channels And Definitions	4-6
4.3 The Attributes Of A Virtual Channel	4-6
4.3.1 Identifiers	4-7
4.3.2 Result Type	4-8
4.3.3 Virtual Function	4-9
4.3.4 Destiny	4-15
4.3.5 Extreme Limits	4-16
4.3.6 Format	4-17
4.4 The RPN Calculator	4-18
4.5 The Virtual Channel Scan Sequence	4-25
4.6 Screen Definitions	4-25
4.7 Contents Of "CONFIG.MON" File	4-26
5.0 THE DEFAULT SOLAR SYSTEM CONFIGURATION	5-1
5.1 Solar System Schematic And Modules	5-1
5.2 Virtual Channel Assignments	5-2
5.2.1 Collector Module	5-3
5.2.2 Heat Exchanger Module	5-5
5.2.3 Piping Module	5-6
5.2.4 Storage Module	5-7
5.2.5 Controller Module	5-8
5.2.6 Load/Energy Module	5-9
5.3 The Default CONFIG.MON and Screen files	5-10

TABLE OF CONTENTS (CONT'D)

Description	Page
6.0 SENSOR CONNECTIONS AND PLACEMENT	6-1
6.1 HP3421A Channel Arrangement	6-1
6.2 Thermocouples	6-3
6.3 Pyranometers	6-4
6.4 Flowmeters	6-4
6.5 Pressure Sensors	6-5
6.6 Event Sensing	6-5

APPENDICES:

A: ERROR MESSAGES

B: USING VEDIT - THE FULL SCREEN EDITOR

C: USING MICRO WATSUN (IBM PC)

CHAPTER 1

INTRODUCTION

The Solar Commissioning Package (SCP) is a portable computer based data acquisition system designed to assist with the commissioning and diagnosis of solar thermal systems.

The SCP is based on an IBM Personal Computer and an Hewlett-Packard HP3421A data acquisition unit with the associated support components.

This "intelligent" data collection system has the ability to measure all key solar performance parameters on a periodic basis, and to automatically average or integrate the readings and save the results on disk at a user selected save interval (usually one hour). The SCP automatically computes "real" module and system performance indices based on these measurements while monitoring. Examples include collector power and energy output, heat exchanger effectiveness, and component efficiencies. Through user-defined formulae, theoretical or expected module performance may also be calculated and compared against the real performance. The SCP will automatically log deviations that exceed user-specified tolerances (e.g., +/-20%) in a "diagnostic-warning" file for later review.

As presently configured, the SCP exists as a "bench-tested" system only. Eventually the hardware will be housed in rugged portable cases, and significant effort will be placed on streamlining the use of the package including the acquisition of "clamp-on" temperature sensors.

The system software also only exists in a bench-tested version. The heart of the software, written in IBM Pascal, is a general purpose monitoring program. All sensors to be scanned, on-line computations, and graphic output are defined

by the user in a set of "configuration" files. the eventual intent with the software is to provide a series of pre-defined configuration files and an easy-to-use interactive front-end to allow on-site configuration. This will allow users to select a "run-disk" preconfigured for the exact system being monitored. System to system changes such as number and placement of heat exchangers, and the corresponding energy/performance formulae would be predefined. The user would then merely choose the correct disk, enter site specific parameters such as collector area and storage size, and leave the system to collect data. The current version of tne SCP contains untested configuration files for an antifreeze DHW system with a counterflow external heat exchanger on the collection side.

The SCP is not intended for "general" use at the present stage of development. Only those with a basic knowledge of computer hardware and software, and general instrumentation techniques should attemp to use the package. The initial field trials and subsequent development phases will remove many of tne hardware and software bugs that are sure to be discovered.

A more detailed SCP reference manual is availaole for tne system. That manual contains more detailed documentation on the operation of tne SCP, and is intended for use by programmers and engineers with a working knowledge of computerized data acquisition systems. The reference manual contains a brief description of the internal operation of the program, source code listings, and numerous notes on the implementation of the system.

For assistance with the SCP, users should contact the Engineering Section of Solar Programs Office, Public Works Canada (613) 998-9817.

CHAPTER 2

SYSTEM HARDWARE SET-UP

This section of the manual is intended as a guide for those installing the SCP computer/data acquisition hardware for the first time. It covers necessary components, cabling, power supply concerns and basic system hardware functional checks.

2.1 System Components.

The SCP is composed of the following:

1. IBM Personal Computer with a minimum of;
    1. IBM DOS 2.00 or 2.10 operating system.
    2. Keyboard.
    3. Memory: 192K Bytes of RAM.
    4. One (preferably 2) 360K DS/DD disk drives.
    5. Colour graphics card.
    6. Video (monochrome) monitor.
    7. Tecmar IEEE card for IBM PC.
    8. Battery backed-up real time date/clock card (a Quadram card was used in the original package).

2. HP 3421A mainframe with three option 020 10-channel multiplexer assemblies and the HP-IB (IEEE-488) bus option.

3. Connecting cables for the video monitor (RCA type), the IEEE bus (> 1m in length), and power cables.

4. Sensors including type-T thermocouples, flowmeters,

pyranometers and status indicators.

2.2 Cable Connections.

The power cables for the IBM PC and the HP 3421A are grounded (3-wire) 120 VAC standard electrical equipment connectors. Note that the HP cable is light coloured while the IBM PC power cable is black. Do not interchange the cables.

The IBM PC keyboard has its own DIN connector which plugs into the DIN socket marked "keyboard" on the back of the IBM chassis.

The video monitor must accommodate a standard composite video signal and an RCA coaxial connector. The RCA/RCA cable plugs into the RCA female plug on the IBM colour graphics card and into the back of the video monitor.

The IEEE cable connects the Tecmar interface card in the IBM PC to the HP 3421A. Note that this cable is attached at either end with knurled machine bolts.

If a printer is connected to the system it can either use the serial (RS-232) or parallel port provide on the QUADRAM board.

Sensor connections to the HP 3421A are discussed in another section.

2.3 Power Source Concerns.

The SCP is designed to automatically boot whenever power is applied to the IBM. No doubt that many of the commercial or industrial envronments where the SCP will be used will have very "unclean" 120 VAC power. Power line transients caused by inductive loading or large current draws could cause problems.

Although the IBM PC power supply design is well thought out, these power problems may render periodic system failures.

The following general guidelines should be observed where practical:

1. Avoid the use of high-impedance extension cords/connectors where possible.

2. Ensure a "solid" hydro ground is connected to the power supply.

3. Attempt to draw power from as close to the true source as possible. This usually means drawing the 120 VAC power from as physically close as possible to the main 230/550 VAC building distribution panel.

4. Avoid sharing power circuits with other large loads such as air conditioners, drill presses, motor circuits, etc.

If power transients are assumed to cause spurious failures, a constant-voltage regulating transformer or other power conditioning means may be a practical solution.

2.4 Basic Hardware Functional Checks.

After all hardware connections are made, the basic operation of each component may be tested.

IBM PC

Insert a system disk [use the "set-up" disk if possible] into the leftmost drive (A:) of the IBM PC and close the drive door. Turn on the video monitor and ensure the brightness is turned up. Flip the red toggle switch on the IBM PC [on right hand side near the back] to the on position [Up is on].

On power up the IBM PC performs a series of internal checks (e.g., memory parity). Between 24 and 45 seconds will elapse while these tests are underway. If any errors were detected, a message will be printed on screen, and the IBM DOS manual should be referenced [Chapter 8].

HP 3421A

The initial power-up test for the HP 3421A should be performed with NO sensors connected to the unit.

Watch the LCD display at the left of the HP 3421A front panel and depress the ON/OFF switch.

At first the "ON" underbar indicator will come on by itself for about 1 second. This will be followed by the error underbar and all channel numbers (0-29) coming on for about 1 second. Audible clicks inside the HP 3421A are normal. The HP LCD display will then go off (except the ON underbar) and channel "9" should be displayed for about 1 second. This indicates that the HP 3421A is configured as IEEE device number 9.

At this point the entire display should go off except the ON underbar. If the error underbar is still displayed it indicates the HP 3421A detected an internal error. If an error was detected record the number that is displayed on the LCD screen and refer to the HP 3421A manual to determine tne problem.

2.5 Environmental Concerns

The environmental restrictions of the IBM PC are more stringent than those for the HP 3421A. The following table is based on the IBM PC limitations:

		SYSTEM ON	SYSTEM OFF
Temperature	Maximum	32 C	43 C
	Minimum	16 C	10 C
Relative Humidity	Maximum	80 %	80 %
	Minimum	8 %	20 %

To maintain the HP 3421A accuracy specifications, HP suggest the an operating temperature within 5 C of the temperature at whicn it was calibrated. The HP 3421A itself can be operated at reduced accuracy specifications over the temperature range of 0 C to 55 C.

2.6 Calibration

The SCP package must be calibrated periodically to ensure the information obtained is reliable and accurate.

Calibration involves checking and adjusting the output of the HP 3421A. The HP 3421A "Operating, Programming and Configuration Manual" contains a brief description of the calibration process on page 20. The HP 3421A Service Manual contains a more elaborate description of the calibration procedure.

It is intended that a menu-driven program will be written after the field trials of the SCP which will allow operator calibration of basic measurements.

As an interim measure, the offset calibration of thermocouples may be accomodated in an approximated manner. The configuration file contains a thermocouple offset value that the MONITOR program will apply to all thermocouple readings. This will enable a constant offset (in degrees C) to be added to or subtracted from all thermocouple temperature readings. This offset should be determined by running the SCP package in the normal manner, and inserting at least one thermocouple in a bath of crushed-ice and distilled water. The mixture should be contained in a vacuum thermos-bottle.

The temperature reading for this thermocouple should (at steady state) be 0.04 C. If a significant deviation is found, the software offset should be altered. This is discussed in more detail later.

CHAPTER 3

SCP SOFTWARE

This section is intended to both introduce and describe the operating characteristics of the SCP software. Note tnat details on configuring the SCP data files for particular applications is covered in another section.

3.1 Overview.

This section describes in general terms the "normal" operation of the SCP software. It assumes the SCP system has already been configured for a particular application, and describes the main functional aspects of the system.

On power-up, the IBM PC performs an internal memory check and proceeds to read and execute a batch file called "AUTOEXEC.BAT". This file normally contains instructions to read the battery backed-up real time clock, delete back-up files on tne diskette, free up any wasted space on the diskette, set up the printer ports and so on. The last instruction contained in the file tells the IBM PC to load and run the monitoring program [called MONITOR].

When MONITOR is run a number of initialization tasks are performed at the beginning. The CONFIG.MON file is read which details the various channel measurements and tne calculations to be performed. The CONFIG.MON file also contains a list (up to 6) of disk file names that the application display screens are contained in.

MONITOR then proceeds to read all screen data into memory one screen at a time. The system displays the actual screens

being read on the video screen as they are loaded.

The program then looks for the "ERROR.MES" and "DIAGS.MES" files on the default disk drive. If one or both are found MONITOR reads the information contained in these files into memory.

The HP 3421A is then initialized witn a "rs" (reset) command. When this command is issued, the HP 3421A begins a power-up sequence, performing various internal checks and opening all multiplexer channels.

Finally the program displays the first screen on the video monitor, and fills all measurement fields on the display with asterisks [asterisks represent an undefined quantity]. MONITOR then begins the monitoring process.

MONITOR will then measure and calculate all channel values, and display these values on the display screens. Every save period (usually 1 hour) all averages (or totals, or integrals as defined for the various channels) are computed and saved on disk. If error or diagnostic conditions are detected, the information is stored in memory until the end of a save period, at which time these messages are also written to disk.

Every midnight MONITOR computes the daily averages, totals or integrals, and saves this summary information to a "DAILY.SUM" file.

Under "normal" SCP operation, the user has no access to the IBM DOS (Disk Operating System) since the program is automatically loaded and run upon power up.

3.2 IBM DOS And DOS Command Summary.

The IBM PC nas its own DOS or Disk Operating System. Under non-dedicated PC applications, the automatic execution of a particular program (such as MONITOR) does not occur. Instead the IBM DOS assumes control after power up.

DOS takes over if the AUTOEXEC.BAT file does not an instruction to run another program as the last statement in the file. The AUTOEXEC.BAT file would still contain commands to read the clock, set up a printer, and so on.

The following DOS description is intended only as a very brief guide. The user should consult the IBM PC DOS manual for more information.

When DOS is waiting for a command, it displays a disk drive letter followed by a wedge:

e.g., A> or B>

The disk drive letter can be any valid disk drive in the system (usually A, B, or possibly C for a RAMDISK). When a string of characters is entered after the prompt, DOS assumes the string to be either a command, or the name of a disk file (on the drive indicated) that is to be loaded and executed.

A>DIR (display directory--DOS command)
or
A>MYPROG (load and run the file called MYPROG.EXE)

Valid File Names

A legal DOS file name has the following format:

D:XXXXXXXX.YYY

Where: D = an optional disk drive [DOS assumes the default drive if omitted]. XXXXXXXX = one to eight characters, YYY = a file extension from one to three characters. Note that any characters entered in lower case are automatically converted to upper case.

A file name can also be the name of a device, such as the console keyboard [CON:], a line printer [LPT1:] or an RS-232 serial communications port [COM1: or COM2:].

The DOS Default Drive

All file operations or specifications of files containing programs to be loaded are assumed to be on the DOS default drive, which is the drive letter displayed in the DOS prompt. To change the default drive enter the new drive letter followed by a colon, and hit <return>.

Brief DOS Command Summary

1. DIR

   Displays the directory of the current (DOS default) drive. DIR X: displays the directory of drive X:.

2. COPY

   Copies one file into another. For example:

   ```
   COPY A:PROG.ONE B:PROG.TWO
         (input)    (output)
   ```

   Creates an exact copy of PROG.ONE (from drive A:) on drive B:, in a file called PROG.TWO. Specifying only a drive name for the output file will copy the input file to that drive and use the same filename as the input file.

   ```
   COPY A:PROG.PAS B:
   ```

3. TYPE/PRINT

   TYPE sends a file to the video display for viewing, and PRINT sends the file to the line printer device [LPT1:]. Note only ASCII/TEXT files can be typed or printed. If the printer is not connected when a PRINT is issued, the DOS may hang (in which case a power-down or control-alt-delete must be done).

   Example:

   ```
   TYPE 04JUN84.DAT
   PRINT B:MYPROG.PAS
   ```

4. DEL or ERASE

Deletes the specified file from the disk. Example:

DEL A:HELLO.PAS, or
ERASE B:05JUN84.DAT

5. RENAME

Changes the name of the specified file. Example:

RENAME OLDNAME.PAS NEWNAME.PAS

6. CHKDSK

Checks the disk drive indicated and frees up any lost space due to non-contiguous block allocation. For example:

CHKDSK B:/F

If lost space was found on the diskette, the /F option, after you are prompted for an answer, will replace these unused areas with special file names called FILEXXXX.CHK, where XXXX is a number. After finishing this command perform a DEL *.CHK command to recover all lost sections.

7. DISKCOPY

Allows an exact copy of a diskette to be made. For example:

DISKCOPY A: B:

DOS will prompt and wait for you to insert the source and target diskettes. If the target diskette has a different format then the source, DOS will automatically reformat the new diskette.

8. FORMAT

Performs a software format of a diskette. This destroys all information on the diskette. All diskettes must be FORMATted before they are used. Note that DISKCOPY automatically formats if necessary. Example:

FORMAT B:

3.3 Disk File Description.

This section explains the various files used and created by the SCP system. The following files are discussed:

1. MONITOR.EXE
2. CONFIG.MON
3. SCREEN FILES [e.g., SCREEN1.SCN..SCREEN6.SCN]
4. SYSTEM.LOG
5. ERROR.MES
6. DIAGS.MES
7. DAILY.SUM
8. DAILY DATA FILES [e.g., 03JUN84.DAT]

The contents of each file and the expectations of the MONITOR program will be discussed.

1. MONITOR.EXE

This is the main monitoring program. To run the program from DOS type "MONITOR<return>". The ".EXE" file extension is used to indicate an executable file.

2. CONFIG.MON

This is the master configuration file for the SCP. The MONITOR program reads this file at the beginning of the run to determine the job title, default MONITOR disk to be used for data storage, the save period, all measured and calculated channel

definitions, and the filenames corresponding to the display screens used by the program. CONFIG.MON must be present on the default DOS disk prior to running MONITOR.

Details on the exact contents of CONFIG.MON and how to alter this file are discussed in another section (see Configuring The Monitoring System).

3. SCREEN FILES

The last several lines of the CONFIG.MON file define the names of all of the screen files to be used with the SCP. The screen files contain comment data (e.g., a schematic of the solar system on one screen, and "exploded" performance views of the major solar system modules on subsequent screens), and a list of channel numbers and their (X, Y) coordinates for display purposes. The channel numbers tell MONITOR which channels are to be displayed on each screen, and the (X, Y) coordinates tell the system where to display them.

Up to six screen files may be defined. In the base solar system (see Section 5), the six screen files are named SCREEN1.SCN to SCREEN6.SCN.

4. SYSTEM.LOG

This file contains a sequential log of system boots and user requested halts. If the SYSTEM.LOG file does not exist on the default MONITOR disk drive [not the DOS default drive], MONITOR creates it, and writes the date/time and an "INITIAL BOOT" message in the file.

Upon reboot (after a power failure), MONITOR writes a "REBOOT" message to the file, along with the date and time.

If the user requests a program halt, a "USER SIGNOFF" message is written to the file.

This file, in ASCII text format, can be TYPEd or PRINTed using the standard DOS commands.

An example of the SYSTEM.LOG file follows:

```
13JUN84 12:38:23 INITIAL BOOT.
13JUN84 13:57:14 USER SIGNOFF.
18JUN84 01:47:48 REBOOT.
```

5. ERROR.MES

This file contains a record of all errors detected by the MONITOR program. The system automatically checks to determine if a particular error message has already been logged. If such a message is found, MONITOR adds one to the occurences count of the error message, and updates the latest date/time the error was detected. Up to 50 different error messages will be recorded. Errors detected after this are ignored.

A list of possible error condition messages appears in an Appendix.

The program automatically reads this file on boot to learn of any pre-power failure errors that were detected. If the file is found, all messages are loaded into memory.

An example of an ERROR.MES file is shown below:

```
6201 HP signal overrange  vc:  3    0 13JUN84 17:04 18JUN84 01:59 1725
7130 RPN divide by zero   vc: 1e    0 13JUN84 10:05 18JUN84 01:59  536
7020 Vc val  abs maximum  vc: 33    0 13JUN84 10:05 18JUN84 01:5e   72
7010 Vc val < abs minimum vc: 59    0 13JUN84 10:06 18JUN84 01:59  135
6037 IEEE card failed     vc: 23    0 13JUN84 10:44 13JUN84 13:40    3
6103 HP pwr on reset det  vc: 53    0 13JUN84 12:46 13JUN84 13:36    3
5000 Pwr failure detected vc:  0    0 18JUN84 01:47 18JUN84 01:47    1
```

One error message is written per line of the file. The first number in the line is the error number. Next follows a 20 character string containing an English description of the error condition. The number printed after "vc:" indicates the virtual channel that was being measured when the error occurred. The number following the VC number is a unique error parameter. This number is used to convey extra information about an error condition. In many cases this number is set to 0. The first date-time and the last date-time the error was detected are printed next. Finally, the last number on the line indicates the number of times the system detected the SAME error.

6. DIAGS.MES

This file contains a summary of all diagnostic conditions detected by the system. A diagnostic condition occurs when the predicted performance of a module is different than the actual performance by more than a preset percentage. A one-line example file is shown below.

```
8500 DIAG test   test  vc:  9    0 03JUN84 23:57 11JUN84 21:59 5507
```

The diagnostic message file is almost identical to the error message file. One diagnostic message is written per line of the file. The first number in the line will always be 8500 (the "error number" corresponding to a diagnostic condition). Next

follows a 20 character string containing the word "DIAG" followed by the names of the two virtual channels that created the diagnostic condition. The number printed after "vc:" indicates the actual virtual channel that is used for this diagnostic warning. The number following the VC number is a unique diagnostic parameter. This number is always set to 0. The first date-time and the last date-time the warning was detected are printed next. Finally, the last number on the line indicates the number of times the system detected the SAME condition.

7. DAILY.SUM

This disk file contains daily summary information for all days the system has run.

One new line is added to the file each day containing the averages, totals or integrals of all channels. An example produced during the SCP bench-test is shown below:

```
JOB: TEST OF SUITCASE MONITORING PACKAGE
SUMMARY
VC NAME  volt V>9? pot resis lnfr tem1  tem2  dtem  tdiag cnts  tlav
03JUN84   9.8 1.00  53 100.2  0.0 19.4  18.8   0.2  1.61   132  18.6
04JUN84   9.8 1.00  70 100.2  0.0 22.6  21.5   0.0  0.23   546  21.0
05JUN84   9.8 1.00  29 100.2  0.0 35.0  35.0  -0.0  0.30   225  35.1
```

8. DAILY DATA FILES

The program creates one file for every day the system operates. The file is always given a file name of the actual date followed by ".DAT". For example:

01JUL84.DAT
29DEC83.DAT

These daily files contain some header information, and one line of data for each save period. If hourly data saves were requested [in CONFIG.MON], 24 lines

of averages/totals are written in the file (one for each hour).

The third last line is a summary line containing the averages/totals for the entire day. It is this line that is also written to the DAILY.SUM file every midnight.

The last two lines of the daily file contain the daily minimum and maximum values for each channel. Note that these values are the extremes found for all scans of each channel. The average values should always be in-between the extremes.

An example daily file is shown below:

JOB: TEST OF SUITCASE MONITORING PACKAGE
04JUN84

VC NAME	volt	V>9?	pot	resis	lnfr	tem1	tem2	dtem	tdiag	cnts	tlav
01:00	9.8	1.00	29	100.2	0.0	36.0	36.0	0.1	0.26	0	36.2
02:00	9.8	1.00	29	100.2	0.0	35.7	35.8	-0.1	0.3?	0	35.8
03:00	9.8	1.00	29	100.2	0.0	35.3	35.3	-0.0	0.15	0	35.4
04:00	9.8	1.00	29	100.2	0.0	35.0	34.9	0.0	0.17	0	35.1
05:00	9.8	1.00	29	100.2	0.0	34.8	34.8	0.0	0.14	0	34.9
06:00	9.8	1.00	29	100.2	0.0	34.6	34.7	-0.1	0.36	0	34.6
07:00	9.8	1.00	29	100.2	0.0	34.4	34.4	-0.0	0.15	0	34.5
08:00	9.8	1.00	29	100.2	0.0	34.1	34.2	-0.0	0.13	0	34.2
09:00	9.8	1.00	29	100.2	0.0	34.0	34.2	-0.2	0.63	25	33.9
10:00	9.8	1.00	29	100.2	0.0	34.1	34.2	-0.1	0.42	0	34.2
11:00	9.8	1.00	29	100.1	0.0	37.0	36.8	0.2	0.50	0	36.9
12:00	9.8	1.00	72	100.2	0.0	20.5	20.6	-0.0	0.39	0	20.5
13:00	9.8	1.00	72	100.2	0.0	20.5	20.6	-0.0	0.39	0	20.5
14:00	9.8	1.00	72	100.2	0.0	20.5	20.6	-0.0	0.39	0	20.5
15:00	9.8	1.00	72	100.2	0.0	20.5	20.6	-0.0	0.39	0	20.5
17:00	9.8	1.00	72	100.2	0.0	20.7	20.7	-0.0	0.16	0	20.7
18:00	9.8	1.00	72	100.2	0.0	20.8	20.9	-0.0	0.19	0	20.8
19:00	9.8	1.00	72	100.2	0.0	21.0	21.0	-0.0	0.16	0	20.9
20:00	9.8	1.00	72	100.2	0.0	21.1	21.1	-0.0	0.26	0	21.1
21:00	9.8	1.00	72	100.2	0.0	21.4	21.3	0.0	0.22	0	21.2
22:00	9.8	1.00	72	100.2	0.0	21.2	21.2	0.0	0.20	0	21.4
23:00	9.8	1.00	72	100.2	0.0	20.5	20.5	-0.0	0.16	22	20.6
00:00	9.8	1.00	55	100.2	0.0	36.6	26.4	-0.0	0.31	499	21.7
AVG/TOTL	9.8	1.00	70	100.2	0.0	22.6	21.5	0.0	0.23	546	21.0
MINIMUMS	9.8	1.00	29	100.2	0.0	17.7	19.8	-0.2	0.00	0	20.2
MAXIMUMS	9.8	1.00	72	100.2	1.0	38.9	36.1	0.1	0.66	205	26.4

3.4 Set-up Vs. Run-Time Operation.

When a non-dedicated diskette is in drive A: (e.g., SET-UP disk or other DOS disk), and power is applied, the IBM will initialize and then display the DOS prompt [e.g., A>]. At this point DOS awaits a command. For the more familiar user, DOS would be used to type or print data files and messages, or re-edit the configuration files.

When the "RUN-TIME" diskette is in drive A:, and the system is powered on, the SCP will boot and begin monitoring. The program may be halted in three ways. First by removing AC power from the IBM computer, second by an operator-requested halt, and third by pressing control-C or control-break twice in a row.

The MONITOR program handles the data measured and calculated using the concept of virtual channels. This concept, fully discussed in the next section, is referenced within this part of the manual since many of the MONITOR program input and output makes direct reference to "VC's". It is sufficient if the reader equate a virtual channel with a true hardware channel for the purposes of this section.

3.4.1 The Run-Time Menu

During run-time, the system displays a brief menu on the last few lines of the screen.

Commands are entered as one-character keystrokes. The keyboard is scanned about once every second. If a key was pressed since the last keyboard scan, the program will attempt to decode the command and take appropriate action.

The commands defined are:

Digit "1" to "9"	: Requests a display screen change.
"D" or "d"	: Display stored data.
"H" or "h"	: Halt program execution.
"?"	: Toggles IEEE diagnostic test flag.
"V" or "v"	: Toggles VC name/number display flag.
"S" or "s"	: Requests display sleep mode.

NOTE

PRESSING CONTROL-C OR CONTROL-BREAK TWICE IN A ROW WILL STOP THE PROGRAM FROM RUNNING.

When a key is pressed, and perhaps a sub-menu is displayed, the system will wait up to several minutes before cancelling the command and continuing the monitoring. This will prevent the system from waiting indefinitely if an operator pushes a key and leaves the site, as would be the case with "normal" menu driven input. A response is normally required within a time period of 30 seconds to 5 minutes, depending on the command being processed.

The function of the commands are discussed below.

1. Digit "1" to "9" (Screen change).

   Entering a digit between 1 and 9 will request that the displayed screen be changed to that screen number. If a digit is pressed that has no corresponding screen definition (e.g., press 5 in a 4 screen system), an "Unknown Command" message will be issued.

   If the digit pressed does correspond to a screen, within several seconds the new screen will appear.

2. "D" or "d" (Display Stored Data).

This command tells the system that the operator wishes to display data on disk. Error messages, diagnostic messages, or measured data may be displayed. After this command is entered, the system will display a sub-menu of display options as follows:

DATA DISPLAY ROUTINE.

1. Examine measurement history.
2. Display diagnostics summary.
3. Display error history.
4. Quit [Continue monitoring].

The system will allow up to 60 seconds between keystrokes before it cancels the command at this level.

1. Examine measurement history.

This command will allow the data measured by the system to be displayed. When "1" is pusned, a sub-sub-menu is displayed as follows:

1. Display days stored in "DAILY.SUM" file.
2. Recall avg/totals for specific day.
3. Recall hourly data for specific day.
4. Return.

The first command, "Display days stored in "DAILY.SUM" file.", will type a list of dates on the screen corresponding to all records in the "DAILY.SUM" file. This list of dates should contain all days that the SCP has monitored a job. Days are displayed in the form "02JUN84". The operator can request this list if uncertain about the data files that exist on disk.

The second command, "Recall avg/totals for specific day.", allows the daily averages and totals to be recalled from disk for a specific day. The "DAILY.SUM" file is read to obtain this information.

The system displays the measured data inside the current set of display screens, at the normal virtual channels (X,Y) output coordinates. The default screen (screen 1) is displayed first. Note the upper right hand corner of the screens will contain the message "STORED DATA" rather than the monitoring message "INSTANTANEOUS SCAN". The time/date for the data recall are also displayed in the upper right hand corner of the screen.

The menu displayed in the bottom window of the screen will now contain the commands "Q" to quit (return to the last menu), or "N" to step to the next record in the file.

The third command, "Recall hourly data for specific day." is similar to the command just discussed. Rather than recall averages and totals for an entire day, this command recalls hourly data for a particular day. A seven character date (e.g., 05DEC81) must be entered to ask the system to search for the correct disk file.

Hourly data is displayed on the screens in the same manner as the daily summary data. The time displayed will be that which was stored with the hourly record on disk.

If the system cannot find a particular file, a message is issued to the operator.

2. Display diagnostics summary.

This command types the diagnostic messages on the video screen. The message is printed in the following format (all on one line on the screen):

" 1. DIAG QCREAL QCEXPT [8500] vc: 21
0 03JUN84/11:22 05JUN84/13:44 #: 12"

The first integer is the message number. A 20 character message follows containing an English description of the diagnosis. In this example, the message "DIAG QCREAL QCEXPT" indicates the values of the channels called "QCREAL" and "QCEXPT" were in disagreement by more than the

allowable percentage. "QCREAL" could refer to the actual measured collector energy output while "QCEXPT" could refer to the expected collector array energy output. The number in square brackets [] is the diagnostic "error number". Next the virtual channel number corresponding to the diagnostic channel given, VC number 21 in the example ("vc: 21"). The first date/time and last date/time strings follow the vc number. Finally, the number of occurences of this condition are printed after the # sign. To determine the actual average percentage difference between the two channels the operator would recall the stored data for the time period of interest, and find the VC data corresponding to this diagnostic channel.

3. Display error history.

   This command types the error messages on the video screen. The message is printed in the following format (all on one line on the screen):

   " 1. IEEE card failed [6050] vc: 22
   0 03JUN84/11:22 05JUN84/13:44 #: 12"

   The first integer is the message number. A 20 character message follows the number containing an English description of the problem. The number in square brackets [] is the official error number. Next the virtual channel number being processed when the error occurred is given, VC number 22 in the example ("vc: 22"). The first date/time and last date/time strings follow the vc number. Finally, the number of occurences of this condition are printed after the # sign.

   An Appendix contains a description of the error messages.

4. Quit [Continue monitoring].

   This command returns control back to the main program.

3. "H" or "h" (Halt program execution).

   Typing an "H" in upper or lower case will result in the system asking a question: "QUIT. Are you sure?".

   If a "Y" or "y" is pressed in response, the program will close all files, update the SYSTEM.LOG file with a "USER SIGNOFF" message, and return control to DOS.

   If any other key is pressed the system will return to continue monitoring.

4. "?" (Toggles IEEE diagnostic test flag).

   This command toggles the state of the IEEE software diagnostic flag. If the software diagnostic flag is ON, a "?" turns it OFF, and if OFF, a "?" turns it ON.

   This command is not intended for use by the system operator.

5. "V" or "v" (Toggles VC name/number diaplay flag)

   This command toggles the flag which allows the current VC name and number to be printed in the lower right of the screen as the channels are scanned. It is not intended for use by the operator.

6. "S" or "s" (Requests display sleep mode)

   The data display will automatically go into a "sleep mode" if no keys have been pushed in 20 minutes. This allows an increase in measurement frequency of some 20% over that if the display is always updated.

   Pushing "S" will put the display into the sleep mode immediately. Sleep mode will remain until any key is pressed.

CHAPTER 4

CONFIGURING THE MONITORING SYSTEM

This section is intended to describe the procedure for configuring the SCP monitoring package. Details on the operation of the SCP MONITOR program and necessary data files are presented. This section must be read prior to altering the default configuration files and screen definitions.

4.1 Overview And System Generic Capabilities

The heart of the SCP is the MONITOR program. MONITOR is written in the Pascal language, and is a completely general purpose monitoring program, not at all specific to solar energy system monitoring. MONITOR could be used to monitor the performance of a building, collect weather data, record specific experiments, and so on, with no changes to the program itself. Earlier versions of the same program written in BASIC have been used to monitor greenhouse energy performance and residential air-to-air heat exchanger installations. The current version of MONITOR has been significantly extended beyond the early BASIC versions.

All functions performed by MONITOR are dictated by the input data files. This includes the measurements to be made, the sensor types and calibration curves, the formulae to be evaluated, and the information to be displayed on the screen.

Automatic Boot Capability

The SCP package handles power failure through the auto-boot feature built in to the IBM PC. If the power fails (120 VAC), the system will cease monitoring. Upon the restoration of line power, the IBM PC loads the current date and time from the real-time clock, and runs the MONITOR program. MONITOR then loads the CONFIG.MON file and the error and diagnostic messages from disk, and continues monitoring. Data is lost for the duration of the power failure.

Basic Measurement Capability

MONITOR is capable of measuring the following signals/sensors, and performing the the following tasks:

1. DC Voltage
2. DC Resistance
3. AC Voltage
4. Type-T Thermocouples
5. Frequency
6. Pulse Counting
7. On-line mathematical calculations

MONITOR can be directed to measure or calculate a quantity, display the result on the video screen(s), average or totalize the scan readings, and store the averages/totals on disk every save period.

Note that the system will measure the output of almost any sensor or transducer. Most sensors output a DC voltage, a resistance or a frequency that varies with the quantity being measured. Since the system can measure all of these basic quantities, most sensors are easily accomodated with no hardware or software changes.

Calculation Capability

The most powerful feature of the MONITOR program is the on-line calculation facility. The RPN calculator function allows the user to define almost any formula, and evaluate the formula as if it was just another measurement function. MONITOR makes no distinction between the results of a measurement function (e.g., thermocouple) and the results of a calculation. They are both treated as a data quantity to be

averaged, stored, displayed and so on. This feature allows the "application" calculations to be altered without the need to change and re-compile the program itself.

The RPN calculator feature allows the measurement of virtually any sensor with the system. Since the basic measurement capabilities include the most common types of signals (e.g., voltage, resistance, etc.), the output of a sensor can be measured and converted into engineering units on-line. For example, if a relative humidity sensor had a DC voltage output of 10 mV per percent, a channel would be defined as the measurement of that DC voltage followed by a conversion calculation. The calculation would divide the DC voltage by 0.01 to obtain the relative humidity in percent. This "channel" then corresponds to the relative humidity directly in percent, and can easily be used in subsequent calculations, stored on disk, or displayed on the screen.

Data Display Capability

The display capability of the MONITOR program is also quite powerful. Conventional data acquisition systems usually record measurements and allow the data to be displayed as a series of tabular outputs with no immediately obvious relation to the physical system being measured.

The MONITOR program allows the the definition of a series of "display screens". The display screens contain both static and changing data. The static data includes all non-changing information such as the schematic representation of the system being measured, data quantity descriptors, and measurement units. A list of channels may be defined for each of the screens. This list consists of a channel number and an (X,Y) co-ordinate pair. If a particular static screen is displayed, MONITOR will display the values of those channels defined in the list at the indicated (X,Y) co-ordinates every scan. This allows the measured or calculated outputs to be viewed instantly, and also allows the viewer to immediately associate the data quantity with the physical system. Even persons unfamiliar with the physical system being measured could obtain a basic understanding of the operation of the system by watching the display screen for several minutes.

Data Reduction

MONITOR can be directed to perform a number of "operations" on the data it acquires. For example, overall averages, totals, or time integrals of the data quantities can be specified.

With the built-in calculation capability, the program can display both primary and secondary data quantities on-line. Temperatures or mass flow rates are examples of primary quantities. Power, energy or efficiencies are examples of secondary quantities.

Since the user specifies the formulae to be evaluated, system performance parameters may be converted into meaningful secondary parameters on-line. This allows a large portion of the conventional system performance analysis to be performed immediately, making results available to the user within seconds of the measurements. More conventional data acquistion systems require the experimenter to write complicated data reduction, display, and analysis routines, which in may cases end up costing more than the data acquistion program itself. After these programs are available, closer examination of the data and development of secondary performance indices usually leave questions unanswered about the system. At that point, a re-run of the experiment is usually out of the question. The on-line computation ability allows much of this analysis to be undertaken as the experiment is underway.

Diagnostic Capability

An on-line diagnostic capability allows any two quantities to be compared every scan, and if the difference exceeds a pre-set percentage, a diagnostic warning is logged. This feature allows, for example, the comparison of measured collector array energy output with an on-line calculation of "expected" collector array output based on a user-input formula.

The diagnostic capability allows the computer to automatically log conditions worthy of further engineering analysis. The SCP package is designed to both collect data unattended, and also display measured results immediately. After a week of unattended system measurement, the designer may review the diagnostic output of the SCP. Potential underperforming components would be clearly flagged in the diagnostic file [e.g., HX effectiveness average disagreement with design

condition : 50 %. Out of range 1000 times.]. Since the diagnosis is also performed on-line, the reasons for the apparent malfunction may be sought immediately. For example, if a heat exchanger was underperforming, the temperature/flow "inputs" to the heat exchanger could first of all be checked for proper value. If acceptable, the manufacturers specification sheet on the product would then be consulted to determine if the component itself was not up to specification, or if the system design around the component was insufficient to guarantee full component performance.

Storage Capability

The MONITOR program automatically stores the results of the measurements and computations at a user-selected save interval.

The data files produced by MONITOR are all ASCII text files. This allows a data set to be printed on the printer, or edited by the user with ease. Numerous software packages for data acquistion store the information in "binary" files. Binary files are readable only by the machine itself and are unprintable. Although more compact than ASCII files, the mass storage devices of today have made binary files a thing of the past. A file unreadable by the operator of the system is of little use and usually leads to data integrity problems and long system debug times.

MONITOR allows the operator to recall stored data from the main command menu. Although version 1.5 of program will cease measurements during this data recall period, relatively minor modifications to the program would remove this limitation.

The user can issue the "display stored data" command from the main menu. MONITOR will then prompt for the specific data to be recalled. Error messages, diagnostic messages and actual stored data may be displayed. If the stored data option is selected, MONITOR asks the user for the date of the data to be displayed (e.g., 02JUN84). MONITOR will then load the desired data from disk (if it exists), and display the information on the system display screens. This allows the operator to recall a historical "picture" of the system performance for any hour or day that measurements were made.

This feature is essential for fast data analysis turnaround. If a particular problem with the system has been detected, the

operator may look at historical performance "pictures" of the system in an attempt to understand the reasons that the performance is low.

4.2 Virtual Channels And Definitions

The monitoring concept revolves around the "virtual channel". A virtual channel, or VC, is a data quantity that MONITOR is instructed to periodically obtain, display, average and save on disk. The VC can be a direct measurement such as a thermocouple, or the result of a series of mathematical operations on other virtual channels.

The VC concept removes the direct relationship between "channels" and software associated with conventional data acquistion systems. It allows the program to be almost application independent. Program alterations are in most cases not necessary when the application of the program is changed. A new application involves the user altering the input data file which contains the VC definitions, re-defining a set of display screens for the application (e.g., schematic information, etc.), and re-running MONITOR.

Each VC must be defined before MONITOR can proceed to acquire, display, and store the values. The VC definition is contained in a set of fixed VC attributes. These attributes tell the MONITOR program how to derive the VC value, what to do with the scan values every save period, the legal range for the VC, how to format the result on disk and on the display, and so on. The VC definitions are contained in the CONFIG.MON file. MONITOR will read this file on initialization to load the definitions into memory.

4.3 The Attributes Of A Virtual Channel

A virtual channel has both fixed (non-changing) and variable attributes. The fixed attributes are defined by the user while the variable attributes are defined by the MONITOR program at run-time.

Examples of fixed attributes include the result type (i.e., a

real number, an integer number or a boolean condition), the function to be used to determine the value (e.g., thermocouple function, DC voltage function, etc.), and the operation to be performed on the results of the function (e.g., overall average, total, time-integral, etc.).

Examples of variable attributes include the value of the VC last scan (e.g., the temperature of a storage tank last scan), the maximum scan value detected so far today, the last time the VC was measured, and so on.

This section introduces the VC attributes.

4.3.1 Identifiers

Each virtual channel has two identifiers associated with it. These identifiers allow the program and the user to associate a data value with a particular virtual channel.

The first identifier is the VC number. This is no more than an index used to reference the VC. All program operations on a VC use the VC number as a reference. Version 1.5 of the program can support up to 64 virtual channels (numbered 1..64).

Each virtual channel can also have a "name". The name is from one to seven characters that the user can use to identify a particular data value. The names used are intended to convey significance about the data value to the user. An example would be the definition of a VC name as "TAIR". This name would logically be connected to the temperature of the air in a particular physical system. Another example is "QCOLL". This would likely indicate a "COLLected" heat flow for a system. Perhaps the energy collected by a solar collector.

When MONITOR creates a daily VC record, the names of all VC's are automatically written as one of the top lines in the file. Data values (e.g., each hour), are appended to this daily file. The VC values are formatted in a manner defined by the user, and written directly under the VC names. This allows the user to immediately associate a particular column with the parameter in the physical system.

The VC names are limited to a maximum of seven characters, or the length of the VC value storage format in characters,

whichever is smaller (see format section later).

VC Fields

Each virtual channel has four fields. Although virtual channel numbers are referenced by an integer, the field witnin a VC that is to be recalled is determined by the fractional portion of the VC number. As a result, 3.0 or 3 selects the zeroth field of VC number 3, 3.1 the first field, and 3.2, 3.3 the second and third fields respectively.

The zeroth field always selects the last scan value of the VC. In the case of a thermocouple, the last temperature measured would be selected witn field.

The third field always recalls the time interval of the last scan. This corresponds to the elapsed time in seconds between the last two measurements on that channel.

The first and second fields select different quantities depending upon the destiny of the VC. The first field selects the destiny operation result performed from the start of this save period up to the current time in the save period. In the case of an overall temperature average, the first field would correspond to the moving average within the save period. If pulses were being totalized, the first field would correspond to the total pulses received during the save period.

The second field also depends on the destiny. It corresponds to the result of the destiny operation performed from midnight last up to the current time of the day. In the case of an overall temperature average, the second field would correspond to the moving average today. If pulses were being totalized, the second field would correspond to the total pulses received so far today.

4.3.2 Result Type

The result type of a virtual channel refers to the actual data type. A VC can be a real number, an integer or a boolean condition.

The result type in no way affects the internal representation or format of a virtual channels value. All VC values are

internally represented as real numbers. The result type boolean can assume either a TRUE or FALSE condition. Internally, TRUE corresponds to 1.0, and FALSE corresponds to 0.0.

Boolean VC's are used for the indication of device status (e.g., pumps, solar system ON/OFF, etc.).

4.3.3 Virtual Function

The virtual function attribute of a VC is an integer that represents the function to be used to determine the VC value. Examples of functions include "thermocouple", "DC voltage", "frequency", or "calculator". MONITOR will invoke the measurement subroutine corresponding to the virtual function requested.

The specification of just a virtual function is not always sufficient to completely determine the value of a VC. Some measurement functions require additional information. If measuring a type-T thermocouple for example, the actual hardware channel that the thermocouple is connected to must also be known.

As a result, four "Function-specific" measurement parameters are also attributes of a virtual channel. These four parameters are integers that contain information in addition to the virtual function number to allow MONITOR to the value of a VC. The meaning of the four numbers is dependent on the function specified.

The virtual function index can range from 1 to 17. The four measurement parameters are integers. The following table outlines the definitions. A blank in the measurement parameter column indicates the value is not used.

Virtual Funct. Number	Function Description	Measurement Parameters 1	2	3	4
1	DC Voltage(autorange)	channel			
3	Two-wire resistance	channel			
4	Four-wire resistance	low-channel(high chan=low+10)			
5	AC voltage	channel			
6	Frequency	channel			
9	Pulse Counter (D/A)	channel	mV/count	max-counts	
10	Type-T thermocouple	channel			
11	HP Reference temp	channel			
13	RTD	low-channel(high chan=low+10)			
14	Diagnostic function	VC1	VC2	percent field	
15	Null function				
16	RPN calculator				
17	Boolean transition	VC			

The first measurement parameter is usually the channel, as indicated above. This refers to the analog hardware cnannel on the HP 3421A.

HP 3421A channels range from 0 to 29. Not all of these channels are usable for input measurements. The HP 3421A comes off-the-shelf with channel 0 and 1 on each 10-channel multiplexer board wired as outputs, and channel 2 on the assembly wired as an attenuated input. Jumpers have been installed in the HP 3421A so that only the third board remains in the off-the-snelf state. Channels 0,1,2,10,11,12 have been rewired as "normal" inputs.

Each of tne virtual functions will now be briefly discussed in more detail.

1. DC Voltage

   This function measures the DC voltage on the indicated channel.

   Tne DCV function is set in the autorange mode, and

will allow the HP 3421A to select the most appropriate range for the measurement. Ranges are 300V, 30V, 3V and 0.3V.

2. Two-wire resistance

This function performs a resistance measurement on the desired channel. A two-wire technique is employed.

Resistances up to 30M ohms may be measured.

3. Four-wire resistance

This is the more precise resistance measurement function and is necessary for the measurement of RTD's. The HP 3421A requires that two channels be used for four wire ohms measurements. The "channel" indicated in the above table refers to one of these hardware channels. The HP 3421A assumes the other channel to be the channel specified plus 10. Hence 5,15 or 6,16 would be suitable pairs.

The HP 3421A must be altered from the basic configuration to enable the four-wire resistance measurements. This alteration involves the installation of internal jumpers, and has not been done in the base SCP package.

4. AC voltage

The AC voltage at the specified channel is measured. AC voltages up to 30 VRMS may be measured.

5. Frequency

This function measures the incoming frequency on the indicated channel. The channel is AC coupled and detects zero-crossings. Frequencies from 1 Hz to

10 kHz may be measured.

6. Pulse Counter (D/A)

The pulse counter function is designed for use with integrating type flowmeters such as manufactured by Neptune Meters.

This function assumes the analog voltage present on the specified channel varies in proportion to the number of pulse outputs of the instrument. The function requires three measurement parameters be specified. The first is the hardware channel on which the voltage source is installed. The second is the number of milli-volts (mV) of analog voltage per count of the pulse stream. The third is the maximum number of counts before the output reverts back to 0.

The concept to measure these integrating type pulse-output meters employs an 8-bit digital counter to count the flowmeter pulses. This 8-bit counter will directly drive a digital-to-analog converter, which would yield an analog output voltage in proportion to the count.

Whenever the "present" count is less than the "last" count, the program automatically computes the net count change using modulo arithmetic. Note that if an 8-bit counter is used, a maximum of 255 counts may be detected per sequential scan.

The function also only considers a count change to have occurred if the analog voltage change exceeds 95% of a one count step. This prevents the incorrect apparent measurement of a full-scale count if the analog input voltage drops only slightly due to noise or an imprecise power source.

7. Type-T thermocouple

The HP 3421A supports the measurement and linearization of type-T thermocouples directly. This function measures the thermocouple at the channel

specified, and returns the temperature in degrees C.

The HP 3421A supports temperature measurements over the range -200 to 400 C.

8. HP Reference temperature

This function allows the HP 3421A internal board thermocouple reference temperatures to be determined. One reference sensor exists on each 10-channel multiplexer assembly. The temperature of the board that tne specified channel is on is returned.

9. RTD (100 ohm)

This function allows 100 ohm resistance temperature detectors (RTD) to be measured. The temperature equation installed is a fairly simple one in which the temperature is assumed linear on resistance. The coefficient is assumed at 0.00392 ohms per degree C, and the RTD resistance is assumed as 100 ohms at 0.0 C.

Note that the HP 3421A must be "jumpered" to measure RTD's. The base SCP package does not contain this modification.

10. Diagnostic function

The diagnostic function allows the value of two virtual channel values to compared. The first two measurement parameters contain the numbers of the two VC's. The third measurement parameter is the allowable deviation between the two values in percent. The percentage is based on the first VC specified if non-zero. If the first VC is zero, tne base for the percentage is taken as the second VC specified. If both VC's are 0.0 then the percentage deviation is taken as 0.

The comparison is made on the basis of the absolute value of the actual percentage difference.

The "value" of a virtual channel which specifies this diagnostic function is the actual percentage difference each scan. Normally an average destiny (see below) is specified for a VC with this function so the average percentage deviation is saved on disk.

If the actual deviation exceeds that maximum specified as the third measurement parameter, a diagnostic warning is logged.

11. Null

This measurement function does nothing.

12. RPN

This function selects the RPN calculator function. No measurement parameters are required for this function.

An RPN expression (see subsequent section for a full description of the RPN calculator) must be defined for a virtual channel with this function or an error occurs.

13. Boolean transition

This function may be used to record a boolean transition on another virtual channel. The value will be assigned 1 if the specified VC has changed from the FALSE to the TRUE state (since the last scan).

The result type of the specified VC must be boolean or an error occurs.

The number of ON/OFF cycles of a solar loop pump is

an example of the use of this function.

4.3.4 Destiny

The destiny attribute of a virtual channel indicates the operation to be performed on the data collected each scan the end of every save period.

1. Overall average.

   Indicates a time-weighted overall average of the VC. The VC value is added into the average registers regardless of the value.

2. Non-zero average.

   Similar to overall average above, but the average is evaluated only if the VC value is non-zero. This destiny would be used if the average non-zero value of a signed quantity was required. A reversing flow rate (with negative sign when reversed) would be accommodated with this destiny. Rather than the off-time reducing the average flow rate the average on-time flow will be recorded.

3. Positive average.

   Similar to the non-zero average above but allows the value of a scan to be added into the time-weighted average only if the value is greater than zero. This destiny would be used, for example, if the average ON-time flow rate of collector array fluid was required.

4. Total.

   This destiny will instruct the system to sum all VC scan values during the save period. Examples of the use of the total destiny include pulse counting or

energy accumulation (i.e., the sum of a power times the scan time).

5. Time-integral.

   This destiny yields an approximation to the time-integral of the quantity. Each scan value is multiplied by the scan time (in seconds) and added into the summation register. This destiny could be used to integrate a solar pyranometer output or compute an energy flow based on the time-integral of power.

6. Positive time-integral.

   Same as above except the scan-value/time product is only added into the total if the VC value is greater than zero. This destiny could be used to find the solar radiation incident on a collector array if negative pyranometer voltage outputs in the evening were to be ignored.

MONITOR will sum the appropriate scan values internally depending upon the desired destiny. At the end of the save period, the actual destiny value is calculated and saved on disk.

4.3.5 Extreme Limits

An upper and lower limit must be specified for each virtual channel. These values are used by the program to detect possible hardware or sensor failures.

The lower and upper limits of a channel are assumed to have units equal to the virtual channel itself, which are not always the units of the raw measurement. An example will clarify this point. Assume a VC was defined as the collector loop pressure based on a DC voltage input of 0-5VDC corresponding to 0-750kPa. The VC measurement function would

be specified as DC voltage. An RPN calculator string would also be defined on the channel to multiply the DC voltage by 150 to convert directly to kPa. The resulting value of the VC would then have units of kPa. In this case the lower and upper limits must have units of kPa (not DC volts).

The extreme limits should be selected such that if the VC value falls outside of the range, a sensor or hardware failure is assumed, and an error condition is logged. The system will ignore any measurement or calculation made if an error occurs during the determination.

4.3.6 Format

The format specifier for a VC dictates the manner in which the VC value is displayed on the screens as well the format the VC is saved on disk.

The format specifier is a real number. The integral part of the number always specifies the total number of characters to be used. The fractional part of the number has slightly different meanings depending upon the result type of the VC.

If the VC result type is real, the fractional part of the format specifier is the number of characters that appear to the right of the decimal place. Note that the decimal point itself also consumes one character. This is analogous to the common FORTRAN F-format. If a negative format specifier is entered, E-format is used and the integral part of the format specifies the total character length.

If the VC result type is integer, the fractional part of the format specifier is ignored.

When the result type is boolean (i.e., TRUE, FALSE), the format specifier takes two meanings. For displaying data on the screens as monitoring progresses, a "TRUE" or "FALSE" character string, or portion of will be displayed. The integral part of the format in this case refers to the number of characters to be used (trailing blanks will be added). To allow the entire word "FALSE" to be displayed, at least 5 characters are needed. Specifying the overall average to be taken on a boolean channel will result in a number indicating the fraction of the save period the condition was true. To accomodate saving this information, the boolean format

specifier is interpreted as a "real" number format specifier. Normally, saving on-fractions to the nearest percentage is sufficient (e.g., format specifier of 5.2 will allow both "FALSE" to be printed and saved fractions of the form "00.33").

The format specifier is also used to dictate how daily averages and totals are saved on disk. It is important to ensure tne format width selected will accomodate the LARGEST value of the quantity expected. For example, assume pulses were to be counted on a channel, and the maximum pulses per hour was estimated as 100. After 24hrs, up to 2400 pulses may have been detected. If a format specifier of 3.0 (integer) was used, the 2400 value would be written as "****" since it overflows the field. A format of 4.0 instead would write an hourly scan as " 100", and the daily scan as "2400".

4.4 The RPN Calculator

The MONITOR program contains an on-line calculation capability that is made possible through use of the "RPN calculator" virtual function. The RPN calculator is an implementation of standard Reverse Polish Notation (abbreviated RPN) calculation machines. RPN is a stack-oriented data processing technique that is extremely efficient for computations. RPN is the main feature of the MONITOR program that enables application independence.

MONITOR allows a matnematical expression to be defined for each virtual channel. The expression can be of almost any complexity, supporting all standard "calculator" functions (e.g., +, -, *, /, root, trigonometric, etc.). RPN allows sensors to be calibrated, and secondary quantities such as power or energy to computed on-line with no changes to tne program itself.

The user defines a character string when a calculation is required. This string contains a sequence of characters that describe to RPN the exact formula that is to be evaluated. Note that the calculation "virtual function" is not distinguished from the normal measurement routines. As far as the MONITOR program is concerned, RPN is just another routine called to determine the value of a virtual channel.

In principle, the RPN calculator operates analogous to

standard RPN calculators (e.g., HP calculators). RPN recognizes two fundamental items: operands and operators. An operand is the data (number) to be operated upon. RPN will allow either numerical constants or the value of a virtual channel to be used as operands. Operators dictate the operation to be performed on the data elements. Examples include "+" for addition, or "sin" for the sine of an angle.

Stack Operation

The RPN calculator operates using the concept of a data stack. The stack is a vertical stack of numbers or operands. When RPN detects a numerical quantity (an operand), the value is immediately placed on the top of the stack. The stack operates as a data storage device with the characteristic that the last data item put on the stack is the first data item removed.

The following analogy may help those unfamiliar with the stack concept. Consider the stack as a long vertical cylinder with the bottom end sealed. This cylinder is used to store data items. Whenever a number is encountered it is thrown into the cylinder. Numbers continue to stack-up inside the cylinder as more data is processed. When an operator such as + or - is encountered, the correct number of data items are pulled from the top of the stack and operated on. The result is then placed back on top of the stack. This process continues until the entire formula has been digested. At this point, only one data item should remain on the stack -- the result of the formula.

Operators

Operators dictate the operation to be performed on the data. For example "+" means: remove the top two numbers on the stack, add them elements together, and place the resulting sum on the top of the stack. "TAN" means replace the top item on the stack with the tangent of the item.

Operators can be unary or binary. Unary operators require only one piece of data to operate. Binary operators require two data items to operate. Examples of unary operators are "TAN", "RUT" (square root), or "ABS" (absolute value). Binary operators include "+", "-", "*", or "/" for addition, subtraction, multiplication, or division.

When a unary operator is encountered, at least one data item must be present on the stack or an error occurs. When a binary operator is encountered, at least two data item must be present on the stack or an error also occurs.
Binary operators operate on the data in the order in which it was placed on the stack. For example, assume "D1", "D2", and "D3" are data items. The RPN sequences below illustrate the order of operations.

RPN SEQUENCE	MEANING
D1 D2 +	D1 + D2
D1 D2 /	D1/D2
D1 D2 * D3 +	D1 * D2 + D3
D1 TAN D2 *	TAN(D1) * D2
D1 D2 ABS SIN -	D1 - SIN(ABS(D2))

The following operators are defined:

Unary:

ABS	Absolute value.
RUT	Square root.
SQR	Square.
EXP	e to the power x.
LN	Natural logarithm.
LOG	Logarithm to base 10.
INT	Truncation.
SIN	Sin (radians).
COS	Cos (radians).
TAN	Tan (radians).
ATN	Arctangent.

Binary:

+	Addition.
-	Subtraction.
*	Multiplication.
/	Division.
POW	Power.
MAX	Maximum of two values.
MIN	Minimum of two values.
>	Logical greater than [Note 1].
<	Logical less than [Note 1].
=	Logical equality [Note 1].
>=	Logical greater than or equal [Note 1].
<=	Logical less than or equal [Note 1].

NOTE

OPERATORS MUST BE IN UPPER CASE IN THE RPN STRING.

Logical operators compare the top two items on the stack. If the result of the comparison is TRUE, the top two items on the stack are replaced with a 1.0. If FALSE, the top two items on the stack are replaced with a 0.0. For example the string "2 1 >" would result in the number 1 on top of the stack since 2 is > 1.

RPN Data Items

Data items may be either constants or virtual channel references. Virtual channel value references are specified by a number sign preceding the VC reference number. Constants are specified by the string representation of the number. E-format is allowed, as are negative signs.

Some examples follow.

Data Item	Meaning
"6.2"	The number 6.2
"1.0000"	The number 1. [Leading/trailing zeros ignored]
"#1"	The zeroth field of VC number 1 (last scan value)
"#1.0"	The zeroth field of VC number 1 (last scan value)
"#22.2"	The second field of VC number 22
"#54.1"	The first field of VC number 54
"1.0e-2"	The number 1.0E-2 or 0.01
"-2.4"	The number -2.4
"- 3.3"	An operator (-) followed by the number 3.3
"-2 RUT"	Error, square root of a negative number.
"7 sin"	Error, sin is lower case not UPPER CASE.

RPN Errors

The RPN calculator may detect numerous error conditions as a formula is being processed. The three basic types of errors are:

1. Bad formula.

   Unrecognizable strings, undefined channel numbers, or incorrect operator sequence may cause this type of error. For example, the sub-string "2.2a" would cause an "Illegal numerical constant" error. Requesting an non-existent virtual channel will result in a similar error message. Defining a formula which either attempts to remove more data from the stack than exists or leaves more than one number on the stack at the end of the formula is also an error condition. For example, the sequence "2 2 + +" would generate a "Binary operator but less than two operands" error message. The sequence "2 3 4 -" would leave two numbers on the stack at the end of the formula (2 and -1). RPN would issue an

"Insufficient operators" error message in this case.

Specifying the calculator virtual function when no formula is defined for that channel will result in an "RPN 0 char in form" error message.

2. Stack Overflow.

The RPN stack can accept at most 20 data items. If more than 20 numbers are pushed onto the stack, an error condition will result.

3. Undefined mathematical operations.

The square root of a negative number, division by zero, or attempting to raise e to a number greater than 70 would lead to undefined operations. RPN will issue error messages when these conditions occur.

4. Undefined VC value.

If the value of a VC is undefined (e.g., thermocouple but out of range), RPN will leave the dependent also undefined. No error message will be logged on disk if this condition occurs.

Formula Format

The MONITOR program has one large array reserved for the storage of all formulae. This is a significant saving in space instead of reserving some 20 or 30 characters each for all VC's. It also allows variable length formulae with little loss of storage efficiency.

RPN uses a space as the element separator. As a result, no spaces can appear in the middle of a element string. For example, "+ 2.2" will be interpreted as the "+" operator followed by the number 2.2. "2. 02" would be interperated as the number 2 followed by the number 2.

Only one space is necessary to separate list elements. Adding more than one will only used up time and memory for no reason.

Some examples of RPN expression results follow.

"3 4 5 + +" will result in 3 + (4+5) = 12
"4 RUT" will result in the square root of 4, or 2
"1.1 SQR" will result in 1.1 squared, or 1.21
"3.14159 SIN" will result in the SIN of pi, 0.0
"2 3 SQR 4 + /" will result in 2/(3**2 + 4) = 2/13

RPN And Measurements On The Same VC

If an RPN expression is defined on a particular virtual channel, it is evaluated after any measurement functions have been executed.

For example, assume VC 7 was defined to be a DCV measurement on analog channel 4. Also assume the RPN expression in the same VC definition was "#4 20 *" Then MONITOR will first determine the voltage at the specified channel and enter it into the VC record. When the VC number 4 reference is encountered in the RPN string, the latest value of VC4 will be recalled (in this case the DCV measurement value), and pushed onto the stack. RPN would then multiply VC4 by 20, and return the result as the "measured value". This feature can be used to convert from raw transducer signals into engineering units directly, without the need to define one VC for the voltage and another VC for the engineering quantity.

Boolean Status

The above feature can also be used to determine whether a device is ON or OFF. Assume a DC voltage is present on hardware channel 7 that is less than 4 volts when a motor is ON, and greater than 4 volts when the motor is OFF (i.e., a relay opens or closes).

Assume VC 11 was defined to measure this voltage, and also contained the RPN string: "#11 4 >". The result of the RPN expression would be 1 if the voltage exceeded 4, or 0 if not. Since the MONITOR program represents boolean (TRUE,FALSE) conditions with a 1.0 and a 0.0, this feature can be used to turn a boolean VC either ON or OFF.

4.5 The Virtual Channel Scan Sequence

MONITOR must be told the order in which you wish the VC values to be scanned. This is done by a list of VC numbers placed in the CONFIG.MON file. Up to 128 VC numbers may be placed in the scan sequence under version 1.5 of the program.

If the application contains many VC's, some calculated and some measured, it may be wise to scan all measured parameters first and then evaluate all expressions. This ensures all measurements are performed as closely in time as possible.

If a particular quantity changes more rapidly than the others (e.g., solar radiation due to cloud cover), this channel may easily be scanned more often than the others by defining the appropriate scan sequence.

Assume VC 7 was to be scanned more often than the others (VC 1 to 8 are defined). The normal scan sequence would be "1 2 3 4 5 6 7 8". VC 7 would be scanned more often if the following sequence was defined : "1 7 2 7 3 7 4 7 5 7 6 7 8".

MONITOR reads the list of VC's in the scan sequence until a negative 1 (-1) is encountered. This signifies the end of the scan sequence. Note that not all VC's defined have to be present in the scan sequence. This allows an application to be slightly altered by omitting the VC references in the scan sequence rather than deleting the VC definitions.

4.6 Screen Definitions

MONITOR has the ability to display the measured data on a set of "display screens". Up to six (6) screens may be defined under version 1.5 of the program. The CONFIG.MON file contains the disk file names of all screens to be defined. Screen files are normally designated with a ".SCN" extension in the DOS environment.

Each screen consists of static and variable data. The static data is fixed for each screen, and consists of comments and/or schematic information relevant to the MONITOR program application. Variable data consists of the value of any virtual channel fields. Up to 25 VC field values may be placed on each screen.

The comment or static screen data is easily entered using a full-screen editor, such as the VEDIT program (see Appendix). This allows the user to move around on the screen with the cursor control keys, and adjust screen comment data for the intended application. MONITOR reads in 25 lines of 80 characters per screen as the comment information. The screen files must contain data in this format for proper operation. Any sequence of characters may be defined on each line. The program reads the data in character by character. The screens may also use the extended IBM PC ASCII characters. This extended set includes numerous special characters that can be used for borders, graphics and so on.

The variable data to be placed on each screen is also defined within the screen files. A list of up to 25 virtual channel numbers and (X,Y) co-ordinates may be specified after the 25 lines of comment data. One line is entered for each VC value to be displayed.

The first number on the line is the VC number, and the second and third correspond to the X and Y co-ordinates respectively. The co-ordinates are interperated as COLUMN=X and ROW=Y. The upper left corner of the screen is the position (1,1). X increases to the right, and Y increases downward.

The VC number entered is taken as a real number, where the integral part reflects the VC number, and the fractional part (i.e., .0, .1, .2 or .3) represents the VC field. See the section on VC Identifiers for a description of VC field values.

Several examples follow:

"3 4 5" - place VC 3 field 0 at column 4 row 5
"3.2 7 9" - place VC 3 field 2 at column 7 row 9
"22.1 21 18" - place VC 22 field 1 at column 21 row 18

4.7 Contents Of The CONFIG.MON File

The previous sections have fully discussed the necessary MONITOR input data and VC attributes. The only remaining issue is the manner in which MONITOR expects to find this data stored in the CONFIG.MON file.

The contents of CONFIG.MON are segmented into four major groups:

1. General data
2. VC definitions
3. Scan sequence
4. Screen definitions

NOTE

REFER TO THE DEFAULT CONFIG.MON FILE DISCUSSED IN SECTION 5 AS THIS SECTION IS READ.

General Data

The general data section is the first 5 lines of the file.

The first line contains the "name" of the job. This is taken to be the first 35 characters on the line and is used as the "job-title" for data display and storage purposes.

The first character of the second line is read as the default disk drive in the IBM PC for data storage and recall. All measured data and error/diagnostic messages will be saved and searched for on this drive.

The third line contains a real number as the first sequence of characters containing no spaces. This represents the data save period in hours. MONITOR will perform averages and totals and save a record to disk every save period. Note that save periods less than 0.5 hours are possible but also use considerable disk space.

The first 20 characters of the fourth line of the CONFIG.MON file contains an initialization string that will be transmitted to the HP 3421A upon start-up. This string is intended to set up the HP 3421A for the desired number of digits of resolution, autozero on or off, autoranging on or off and so on. Refer to page 85 of the HP 3421A Operating, Programming, and Configuration Manual for a list of instructions. Note that in version 1.5 of the MONITOR program

this sequence is not acknowledged since the "high-level" HP 3421A commands are employed (e.g., "DCV", "TWO", etc.). Provision was made for this string so that later versions of the program may invoke the advanced HP 3421A command set.

The fifth line in the CONFIG.MON file contains as the first sequence of characters a real number indicating the desired thermocouple offset to be used. This provision was added to allow the thermocouple measurements to be calibrated until the more sophisticated calibration program is developed. The number entered on this line is subtracted from each thermocouple reading. Normally one thermocouple is placed in an ice-bath and the measured temperature entered on this line. If a negative number is entered it will effectively lift the thermocouple output (i.e., double minus equals an addition).

VC Definitions

The next major section of the CONFIG.MON file contains the virtual channel definitions. After reading the first five lines of the file, MONITOR will search for the next line in the file that contains a "$" sign as the first character. The VC definitions are assumed to begin on the line following this "$" sign.

This allows the user to insert various comment and header data at the top of the VC definition table.

After locating this first "$" sign, MONITOR proceeds to read the VC definitions until another "$" sign is encountered as the first character of a line. At this point the VC definitions are assumed to be complete.

Each VC definition must be contained on one line of the file. Comments may also preceed or follow any VC line. MONITOR assumes a valid VC definition line starts with a numeral character (0..9) as the first non-blank character of the line. DO not enter a comment line that begins with a numeral or the program will fail.

The VC definition lines contain all fixed VC attributes as discussed earlier. A free-format read is performed on numbers allowing extra spaces to be added where necessary, and removing the restriction that real numbers must contain decimal points. Some rules must be followed when defining a VC definition line, as illustrated below.

The program looks for data in the following order (all on one line):

1. The VC number (1..64).
2. The VC name (7 contiguous characters).
3. VC result type (an integer: 0=real, 1=integer, 2=boolean).
4. VC measurement destiny (an integer: 0..6).
5. VC virtual function (an integer:1..17).
6. Function specific measurement parameters (4 integers).
7. Lower limit (a real or integer: absolute minimum VC value).
8. Upper limit (a real or integer: absolute maximum VC value).
9. Output format (a real number).
10. RPN string (all characters between the next non-blank and the end of the line).

After all VC definition lines are read (i.e., the next data line in the file contains a "$" as the first character), MONITOR reads away the "$" line and starts searching for the next "$" sign line, which represents the next data group.

Scan Sequence

The VC scan sequence is read next. To allow comment data to be ignored, the program reads away lines until it finds another "$" sign as the first character of the line (as mentioned above). The next line in the file is assumed to contain the start of the VC scan sequence.

The VC scan sequence is a series of integers separated by spaces. The integers are the VC numbers in the scan sequence. For example "1 2 3 4" would specify the first four VC's be

determined in order.

The program continues reading the scan sequence until a -1 is found, which signifies the end of the scan sequence.

Note that the scan sequence will ignore VC's that were not defined above. Up to 128 VC's may be entered in the sequence.

Screen Definitions

The last group of data contained in the CONFIG.MON file is the screen file name block. After reading the VC scan sequence, MONITOR looks for another line beginning with the "$" character.

Up to six disk file names can be defined. The file name is assumed to consume up to the first 14 characters of each line. This allows the screen files to be located on any drive since DOS will only use the DOS default drive when a drive letter is not specified as the first character of the filename. Only one filename per line may be entered. If no drive specifier is included in the file name (e.g., "SCREEN1.SCN" as opposed to "B:SCREEN1.SCN"), MONITOR will look for each screen on the same disk drive that the CONFIG.MON file is located on (the DOS default drive--not to be confused with the MONITOR default drive).

If less than six screens are used only enter the names that exist.

MONITOR continues reading filenames until another "$" sign is detected as the first character of the next line in the file. At this point the information has all been read and the CONFIG.MON file will be closed.

Errors

Due to the size of the MONITOR program, front-end error checking has been kept to a minimum. This occurred since a menu-driven front end is intended to be added to the SCP package to accept data from the keyboard and create the CONFIG.MON file. This front end program will contain all error checking so that the MONITOR program should never receive bad input data.

If MONITOR detects an error while reading the CONFIG.MON file,

an error message will usually be issued. The most common examples are "array subscript out of bounds", or "data format error". An array subscript out of bounds error will occur if, for example, a VC number greater than 64 or less than 1 was entered, or if too many RPN characters were included in the file and the master formula storage vector was too short. The data format error could arise if a "number" contained an illegal character or too few data items were included in the file.

CHAPTER 5

THE DEFAULT SOLAR SYSTEM CONFIGURATION

5.1 Solar System Schematic And Modules

A "default" solar system configuration has already been created to monitor a basic solar system. The CONFIG.MON file, as well as six screen files (SCREEN1.SCN...SCREEN6.SCN) have been created for this base solar system.

NOTE

PRIOR TO READING THIS SECTION REVIEW SECTION 4.0: CONFIGURING THE MONITORING SYSTEM.

The default system is a liquid-base solar heating system with a collection-side counterflow heat exchanger. The solar preheat tank is plumbed in series with the auxiliary tank. A schematic of the default system is shown in Figure 5.1.

If monitoring a solar system different from the default system shown here, the configuration files should be altered to ensure meaningful results are obtained. Many problems can arise if this alteration is not undertaken. For example, the default file expects certain sensors on certain hardware channels. If a particular sensor is not connected to the HP 3421A, the program will still attempt to measure and record the value. In most cases (e.g., thermocouples), a

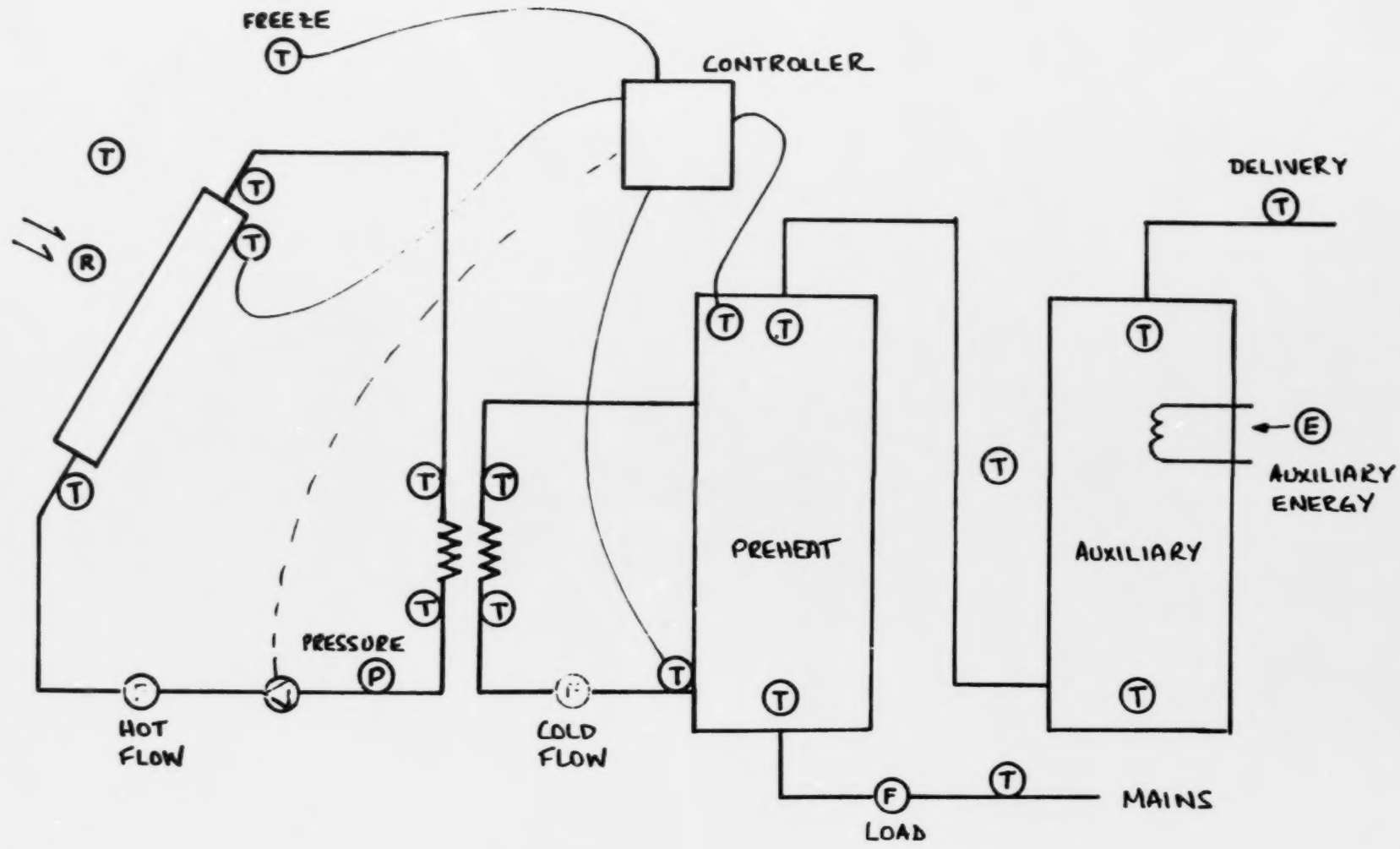


FIGURE 5.1 DEFAULT SOLAR SYSTEM SCHEMATIC

disconnected sensor will usually cause a signal overrange error, which will be caught by the software (asterisks are displayed and stored on disk whenever an error condition is detected on a channel). This overrange indication may not always occur (especially when measuring larger voltages, as in the case of status indicators).

The default system configuration also contains numerous formulae which define energy flow and performance indices. Applying these formulae to different systems may not yield meaningful results. For example, the definition of heat exchanger effectiveness in the default file is the ratio of two temperature differences (counterflow). This can not be directly applied to (say) an immersed coil exchanger.

5.2 Virtual Channel Assignments

The default configuration file contains a list of over 60 virtual channel definitions. Each virtual channel (VC) is either measured, or calculated based on other VC's. Each VC is also assigned a name to help with identification of the data values off-site and in the storage files.

The default system is segmented into major system modules for analysis purposes:

1. Collector array.
2. Heat Exchanger.
3. Pipe system.
4. Storage.
5. Controller.
6. Load/Energy.

Each module is comprised of a series of actual sensor measurements (e.g., temperatures), and module performance equations.

The following tables identify the parameters measured and calculated, and the expected sensor type or output. The user should refer to the actual CONFIG.MON file, later in this

Section, for a more detailed description of the default measurements. This file defines all VC quantities, destinies, formats, and includes comment data relating to the default system.

Site-to-Site Parameter Changes

Some parameters of the solar system, such as collector area, storage volume, or heat exchanger design effectiveness change from site-to-site. The default configuration file contains values for these numbers that will require adjustment each time, even if the basic system design matches the default system.

The CONFIG.MON file contains numerous comments to assist the user with adjustments. During set-up, the user must edit the CONFIG.DEF/.MON file using the full screen editor and alter those values that do not suit the system (see Section 4).

5.2.1 Collector Module

The collector array is treated as a "black-box" with single inlet and outlet temperatures, and a fluid flow rate. The solar radiation and ambient temperatures are considered part of the collector module.

Collector Array Measurements;

Name	Hardware Channel	Description/Default Sensor
IRAD	0	Solar irradiation (W/m2) - Pyranometer 88000 W/m2 per volt
TAM	1	Ambient temperature (C) - Type T thermocouple
TI	2	Array inlet temperature (C) - Type T thermocouple
TO	3	Array outlet temperature (C) - Type T thermocouple
HFTL	4	Total volume through collector (l) - Neptune meter witn D/A output and 1 ppl
PRC	5	Collector loop pressure (kPa) - voltage output sensor @200 kPa/volt

Collector Array Calculations;

Name	Description/Formula Used
EINCMJ	Incident energy on array - IRAD*AREA*TIME, (MJ)
HFR	Collector flow rate - HFTL/scan time, (l/min)
QCREAL	Collected energy - HFTL*0.00419*(TO-TI), (MJ)
DT/I	Array delta-T-over-I - (TI-TAM)/IRAD, (C.m2/W)
QCEXPT	Expected array energy output - (FrTa-FrUl*DT/IRAD)*EINCMJ, (MJ)
PRD	Expected array pressure - constant 350 kPa
DELTA	Array temperature rise - (TO-TI), (C)
REFF	Real array efficiency today - QCREAL*100./EINCMJ, (%)
EEFF	Expected array efficiency today - QCEXPT*100./EINCMJ, (%)

Collector Array Diagnostics;

Name	Channels That Are Compared
DGP	Real vs expected loop pressure - Diagnostic tolerance +/- 20 %
QCD	Actual vs expected array energy - Diagnostic tolerance +/- 20 %

5.2.2 Heat Exchanger

The heat exchanger is assumed to be a counterflow "constant" effectiveness installed on the collection side of the storage.

Heat Exchanger Measurements;

Name	Hardware Channel	Description/Default Sensor
THI	11	Hot fluid inlet temperature - Type T thermocouple, (C)
THO	12	Hot fluid outlet temperature - Type T thermocouple, (C)
TCI	13	Cold fluid inlet temperature - Type T thermocouple, (C)
TCO	14	Cold fluid outlet temperature - Type T thermocouple, (C)
CFTL	15	Total volume through HX cold side - Neptune with 1 ppl D/A output, (1)

Heat Exchanger Calculations;

Name	Description/Formula Used
CFR	Cold side flow rate - CFTL/scan time, (l/min)
DTH	Hot side temperature drop - THI-THO, (C)
DTC	Cold side temperature rise - TCO-TCI, (C)
QHENR	Energy extracted from hot side - HFTL*0.00419*DTH, (MJ)
QCENR	Energy gained by cold side - CFTL*0.00419*DTC, (MJ)
ERE	Real effectiveness - DTC/(THI-TCI)*100., (%)
EEX	Expected effectiveness - constant 75, (%)
NRL	Real efficiency - QCENR/QHENR*100., (%)
NEX	Expected efficiency - constant 90, (%)

Heat Exchanger Diagnostics;

Name	Channels That Are Compared
EDI	Daily average effectiveness: ERE and EEX - Diagnostic tolerance +/- 20 %
NDI	Daily average efficiency: NRE and NEX - Diagnostic tolerance +/- 20 %

5.2.3 Piping System

Piping System Measurements;

Tne piping system is composed of the array inlet-side and outlet-side pipes. The temperatures used to determine actual pipe heat loss are based on the actual array inlet and outlet temperatures (for the collector end), and on the neat exchanger not-inlet and not-outlet ports for the other end. The collector loop flow (HFTL) is used to compute pipe heat loss.

Piping System Calculations;

Name	Description/Formula Used
QPREL	Real piping heat loss - HFTL*0.00419*(TI-THI+TO-THO), (MJ)
QPEXP	Expected piping heat loss - [1.0*(TI-TAM)+1.0*(TO-TAM)]*time/1e6, (MJ)

Piping System Diagnostics;

Name	Channels That Are Compared
QPDI	Daily total pipe losses QPREL vs. QPEXP - Diagnostic tolerance +/- 20 %

5.2.4 Storage

The storage system assumes a preheat tank is plumbed in series with an auxiliary tank. The heat exchanger cold-side fluid warms the storage fluid.

Storage Measurements;

Name	Hardware Channel	Description/Default Sensor
TPT	16	Preheat tank top temperature - Type T thermocouple, (C)
TPB	17	Preheat tank bottom temperature - Type T thermocouple, (C)
TAT	18	Auxiliary tank top temperature - Type T thermocouple, (C)
TAB	19	Auxiliary tank bottom temperature - Type T thermocouple, (C)
TEN	22	Storage surrounding temperature - Type T thermocouple, (C)
TMN	23	Mains temperature - Type T thermocouple, (C)
TDE	24	Water delivery temperature - Type T thermocouple, (C)

Storage Calculations;

Name	Description/Formula Used
TPA	Average preheat tank temperature - (TPT+TPB)/2, (C)
TAA	Average auxiliary tank temperature - (TAT+TAB)/2, (C)

5.2.5 Controller

A standard temperature-differential controller is assumed to control the system. The controller has two control-inputs: collection temperature and storage temperature; and two protection inputs: freeze sensor and an overheating sensor. All four temperatures are measured in the default configuration.

The system is "expected" to be on if the collection-storage control temperature difference exceeds a preset amount (15C), AND if the freeze sensor is greater than 5C AND if the overheating sensor is below 75 C. Otherwise it is expected to be OFF.

Controller Measurements;

Name	Hardware Channel	Description/Default Sensor
TCC	6	Collector control temperature - Type T thermocouple, (C)
TTC	7	Storage control temperature - Type T thermocouple, (C)
TOV	8	Overheat actual temperature - Type T thermocouple, (C)
TFZ	9	Freeze actual temperature - Type T thermocouple, (C)
SYON	10	System ON indicator - TRUE if > 4VDC, FALSE if < 4VDC

Controller Calculations;

Name	Description/Formula Used
SRTM	Solar system run time - sum of time if SYON TRUE, (hrs)
DSTAT	Expected system ON state - Expected on if: (TCC-TTC)>15 AND (TOV<75) AND (TFZ>5)
REXPT	Expected system run time - sum of time DSTAT is TRUE, (hrs)

Controller Diagnostics;

Name	Channels That Are Compared
OND	ON/OFF condition: SYON vs. DSTAT - Diagnostic tolerance +/- 10 %

5.2.6 Load/Energy

The load/energy part of the system determines the total volume of fluid that has passed to the load and the total, auxiliary, and solar energy quantities tnat have been supplied.

Load/Energy Measurements;

Name	Hardware Channel	Description/Default Sensor
LFTL	25	Total load water volume - Neptune with D/A, 1 ppl, (l)
AUXE	26	Auxiliary energy - D/A sensor with 0.5 pulse/MJ, (MJ)

Load/Energy Calculations;

Name	Description/Formula Used
LFRT	Load water flow rate - LFTL/scan time, (l/min)
LENR	Energy to load - LFTL*0.00419*(TDE-TMN), (l/min)
SEAC	Solar energy to load - LENR-AUXE, (l/min)
FSOL	Daily fraction solar - SEAC/LENR*100., (l/min)

5.3 The Default CONFIG.MON and SCREEN Files

Hard copies of the default CONFIG.MON and the six screen files follow. The default CONFIG.MON file is "permanently" stored in a file called "CONFIG.DEF". This ensures the default file is saved regardless of the actual application running.

The screens have been named SCREEN1.SCN through SCREEN6.SCN. The CONFIG.DEF file and the screen files reside on both the "Run-Disk" and the "Set-Up" disk.

```
BASE SCP SOLAR SYSTEM
0       : [ DEFAULT DISK DRIVE FOR DATA SAVE/RECALL ]
1.0     : [ DATA SAVE PERIOD IN HOURS ]
m3 c1 ra1
2.5     : [ THERMOCOUPLE OFFSET AMOUNT (TO CORRECT HP CALIB) -- T modified as Tnew := Told - offset ]


VC VC      RS MES VIR FUNC SPECIFIC LOWER  UPPER  SVE
 # NAME    TP DES FUN MEASUR PARAS  LIMIT  LIMIT  FMT  RPN CALC FORMULA
-- ------- -- --- --- ------------ ------ ------ ---  ------------------
1
 1 IRAD     1  3   1  0 0 0 0       -50    1300  4.0  #1 88000 *
    ..Irradiance in W/m2.  DC voltage measurement.  Analog channel 0. The -50 accounts for negative pyranometer output at night.
      The 88000 is the proportionality constant converting volts into W/m2.  A non-zero average is performed for the
      save period

 2 EINCMJ   0  4  1e  0 0 0 0         0    99    6.1  #1 10 * #2.3 * 1.0e6 /
    ..Energy incident on collector array in MJ.  Based on time-integral of W/m2-area product.  The #2.3 in the RPN string
      recalls the time interval for the last scan of channel 2.  To be totalized every save period.  The 10 in the RPN
      string is the collector area in m2.

 3 TAM      1  1  10  1 0 0 0       -50    50    3.0
    ..Ambient temperature.  Type T thermocouple on analog channel 1. The overall average is to be recorded.

 4 TI       1  1  10  2 0 0 0       -50    250   3.0
    ..Collector Inlet temperature.  Type T thermocouple on analog channel 2. The overall average is to be recorded.

 5 TO       1  1  10  3 0 0 0       -50    250   3.0
    ..Collector Outlet temperature.  Type T thermocouple on analog channel 3. The overall average is to be recorded

 6 HFTL     1  4   0  4 10 255 0     0    255   6.0  #6 1 *
    ..Volume of fluid that passed through the collector last scan in Litres.  A total is to be recorded every save period.
      Channel assumes the analog voltage on hardware channel 4 is proportional to the number of pulses put out by the flowmeter.
      The 1 in the RPN string is the number of litres per pulse.

 7 HFR     -1  3  1e  0 0 0 0         0    200   3.0  #6 60 * #7.3 / #25 *
    ..Collector loop average flow rate for the last scan period in l/minute.  Based on the number of litres that passed through
      the collector divided by the elapsed time as shown in the RPN string.  The 60 in the RPN string converts l/s to l/min.
      The #25 ensures this quantity is only averaged when the system is on.

 8 QCREAL   0  4  1e  0 0 0 0         0    99    6.1  #6 0.00419 * #5 #4 - *
    ..Actual energy collected by the collector array in MJ.  The 0.00419 is the
      specific-heat/density product of the collector fluid in MJ/l.K.

 9 DT/I     0  1  1e  0 0 0 0      -0.2    3.0   5.2  #4 #3 - #1 /
    ..The 'delta-T over I' parameter computed as (TI - TAM)/IRAD. Units C.m2/W.

10 QCEXPT   0  4  1e  0 0 0 0         0    99    6.1  #2 0.7 5.0 #9 * - * #27 *
    ..The expected collected energy in MJ.  The RPN string contains the following numbers : 0.7 is the FrTa. 5.0 is the FrUl.
      Note the #27 in the RPN string zeros the expected energy if the expected ON/OFF state is OFF.

11 QCD      1  1  14  8 10 20 2     -999    999   4.0  #25
    ..Diagnostic channel for collector array performance.  Note the 2 in the measurement parameter sequence dictates the
      comparison of the real/expected collected energy up to this point in the day (to avoid transient problems).
      The 20 in the measurement parameter list is the allowable deviation in percent before a diagnostic condition is logged.

12 PRC      1  1   1  5 0 0 0         0    750   3.0  #12 200 *
    ..Collector loop pressure in kPa.  The channel assumes an analog voltage proportional to pressure is connected to hardware
      channel 5.  The 200 in the RPN string is the conversion factor from volts to kPa for the sensor (assumed linear).
```

```
13 PRD     1  1  16  0 0 0 0          0    750   3.0  350
    .This is a constant channel containing the design pressure for the collector loop. The design pressure is given by the
     number in the RPN string (350 here). Units are kPa.

14 DGP     1  2  14  12 13 20 0     -999    999   3.0  #25
   ..A diagnostic channel for collector loop pressure. Note that this diagnosis is only performed if the system is ON
     (i.e. #25 in RPN). The default acceptable range is +/- 20 % (see 20 in measurement parameters).

15 DELTA   0  1  16  0 0 0 0        -100    100   5.2  #5 #4 -
   ..Array temperature rise in C. Difference of the outlet and inlet temperatures.

16 PEFF    1  1  16  0 0 0 0        -100    100   4.0  #10.2 #2.2 / 100 *
   ..The actual array efficiency. This channel is the division of the energy collected today by the energy incident today.

17 EEFF    1  1  16  0 0 0 0        -100    100   4.0  #10.2 #2.2 / 100 *
   ..The expected efficiency of the array today. The division of the expected collector output by the incident energy so far today.

PIPE SYSTEM

18 QPREL   0  4  16  0 0 0 0        -10     10    6.1  #6 0.00419 * #5 #30 - #31 #4 - + * #18.3 *
   ..Real pipe heat loss in the system was determined by the actual temperature drops.
     Qloss = (volume of fluid passed * 1) * 0.00419 * (total temp drop). The total temp drop is given as COLLout - Hxhotin + Hxhotout -
     COLLin, C. The 0.00419 is the fluid heat-capacity/density product in MJ/l.C.

19 QPEXP   0  4  16  0 0 0 0        -10     10    6.1  1.0 #4 #3 - 1.0 #5 #3 - + #19.3 * 1.0E6 / #27 *
   ..Expected heat loss from piping. The heat loss is based on two temperature differences. The first being the collector inlet - to -
     ambient difference, and the second being the collector outlet - to - ambient difference. The first 1.0 in the RPN
     string is the inlet piping UA value in W/K. The second 1.0 is the outlet piping UA value also in W/K.
     The #27 in the RPN string zeros this virtual channel if the system is not expected to be operating.

20 QPDI    1  1  14  18 19 20 2     -999   999   4.0  #25
   ..Pipe loss diagnostic channel. Compares field 2 of the REAL vs EXPECTED pipe heat loss. Field 2 is the total MJ lost so far
     today. The #25 in the RPN string ensures the comparison is only done when the system is operating.

CONTROLLER

21 TCC     1  1  10  6 0 0 0         -50    250   3.0
   ..Controller actual collection-sensor temperature. This is a type T thermocouple on hardware channel 6.

22 TTC     1  1  10  7 0 0 0         -50     99   3.0
   ..Controller actual collection-sensor temperature. This is a type T thermocouple on hardware channel 7.

23 TOV     1  1  10  8 0 0 0         -50    100   3.0
   ..Controller actual collection-sensor temperature. This is a type T thermocouple on hardware channel 8.

24 TFZ     1  1  10  9 0 0 0         -50    200   3.0
   ..Controller actual collection-sensor temperature. This is a type T thermocouple on hardware channel 9.

25 SYON    2  1   1 10 0 0 0          0     1    5.2  #25 4 >
   ..Solar system ON/OFF indicator. Assumes a DC voltage is present on channel 10. If the voltage
     exceeds 4.0 VDC the system is considered ON. Otherwise it is considered off.

26 SRTM    0  4  16  0 0 0 0          0     1    5.2  #25 #26.3 * 3600 /
   ..Solar system run time in hours. Based on the sum of all scan times when the SYO (system on) channel
     indicates the system is on.

27 DSTAT   2  1  16  0 0 0 0          0     1    5.2  #21 #22 - 15 > #23 75 < #24 5 > * *
   ..Desired State Of solar system. This is the expected ON/OFF condition. The system is expected to be on
     when ALL of the following conditions are true: [TCC (Tcollector-control) - TTC (Ttank-control) > 15.0]  AND
     [TFZ (Tfreeze) > 5.0 C]  AND  [TOV (Toverheat) < 75 C ]. Note the three control numbers are embedded
```

in the RPN string above (i.e., 15, 5 and 75).

28 REXPT 0 4 16 0 0 0 0 0 1 5.2 #27 #28.3 * 3600 /
.Expected System run time in hours. This is the same as SRT above except the desired ON/OFF state is used to determine the scan times that are added.

29 OND 1 1 14 25 28 10 0 -999 999 3.0
..Diagnostic channel for ON/OFF condition of system. Compares the expected state of the system with the actual state. The 10 in the measurement parameter list is the percentage to which these two channels must agree before a diagnostic condition is logged. Since the program represents boolean (ON/OFF) conditions as either a 0 or a 1, the percentage here is almost meaningless. It must be large enough to accomodate the numerical precision only.

HEAT EXCHANGER

30 THI 1 1 10 11 0 0 0 -50 250 3.0
..Heat exchanger hot-fluid-inlet temperature, C. Type T thermocouple on hardware channel 11.

31 THO 1 1 10 12 0 0 0 -50 250 3.0
..Heat exchanger hot-fluid-outlet temperature, C. Type T thermocouple on hardware channel 12.

32 TCI 1 1 10 13 0 0 0 -50 100 3.0
..Heat exchanger cold-fluid-inlet temperature, C. Type T thermocouple on hardware channel 13.

33 TCO 1 1 10 14 0 0 0 -50 100 3.0
..Heat exchanger cold-fluid-outlet temperature, C. Type T thermocouple on hardware channel 14.

34 CFTL 1 4 9 15 10 255 0 0 255 6.0 #34 1 *
..Volume of fluid that passed through the cold side of the heat exchanger last scan in Litres. A total is to be recorded every save period. Channel assumes the analog voltage on hardware channel 15 is proportional to the number of pulses put out by the flowmeter. The 1 in the RPN string is the number of litres per pulse.

35 CFR 1 3 16 0 0 0 0 0 100 3.0 #34 60 * #35.3 /
..Cold HX fluid flow rate in l/min. Based on time derivative of CFT above. The 35.3 recalls the last scan-time for channel 35. The 60 converts l/s to l/min.

36 DTH 0 2 16 0 0 0 0 -30 50 4.1 #30 #31 - #25 *
..Delta-T Hot Side. Given as Thotinlet - Thotoutlet.
The #25 in the RPN string zeros this quantity if the system is not operating.

37 DTC 0 2 16 0 0 0 0 -30 50 4.1 #33 #32 - #25 *
..Delta-T Cold Side. Given as Tcoldoutlet - Tcoldinlet.

38 QHEMP 0 4 16 0 0 0 0 0 2 6.1 #34 0.00419 * #36 *
..Hot side energy flow in MJ. This is the amount of energy that was extracted from the fluid as it flowed through the hot side of the heat exchanger during the last scan period. It is based on the total volume of fluid that passed through the hot side of the exchanger and the temperatures of the inlet/outlet during the scan. The 0.00419 is the density/heat capacity of the fluid in MJ/L.C.

39 QCEMP 0 4 16 0 0 0 0 0 2 6.1 #34 0.00419 * #37 *
..Cold side energy flow in MJ. This is the amount of energy that was added to the cold fluid as it flowed through the cold side of the heat exchanger during the last scan period. It is based on the total volume of fluid that passed through the cold side of the exchanger and the temperatures of the inlet/outlet during the scan. The 0.00419 is the density/heat capacity of the fluid in MJ/L.C.

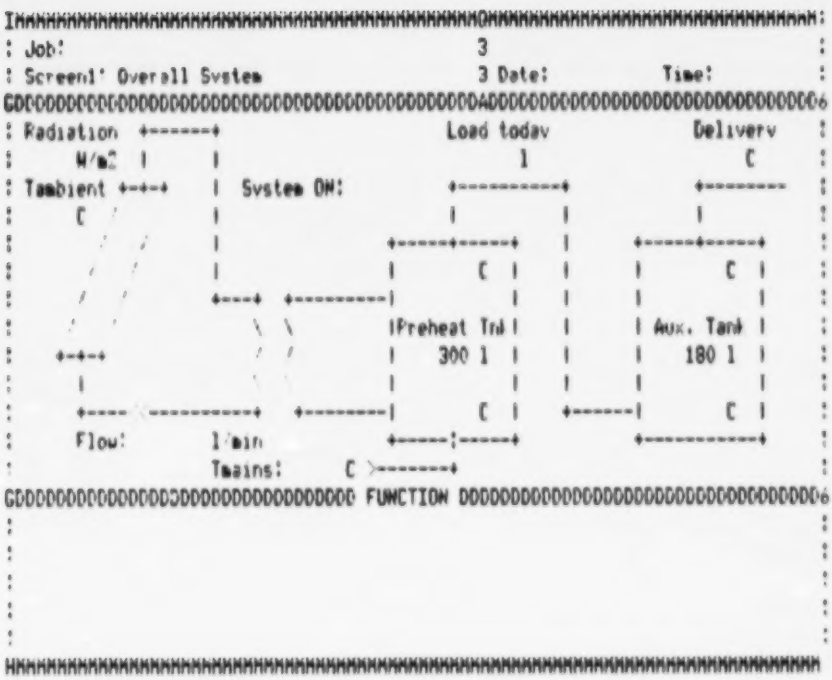
40 ERE 1 2 16 0 0 0 0 -50 300 3.0 #37 #30 #32 - / 100 * #25 *
..This is the actual HX temperature effectiveness based on the measured temperatures. The effectiveness is given as the ratio of cold fluid temperature rise divided by the maximum temperature difference in the heat exchanger (ie Thotinlet - Tcoldinlet). The #25 in the RPN string zeros this quantity if the system is not operating.

41 EEX 1 2 16 0 0 0 0 0 100 3.0 75 #27 *

```
..This is a constant virtual channel that contains the design heat exchanger effectiveness. The number 75 in the RPN string is t
  design value. The #2" in the RPN string zeros this quantity if the system is not operating.

42 HRL     1  2  1e  0 0 0 0        -50    200   3.0  #39.2 #38.2 / 100 *
  ..This is the real efficiency of the heat exchanger. Given as the ratio of cold-fluid-energy-input to hot-fluid-energy-output.
    The energy values are taken as that from midnight until this point in the day (ie daily totals).

43 HEX     1  2  1e  0 0 0 0        -50    100   3.0  90 #25 *
  ..Expected HX efficiency. This is a constant virtual channel to contain the value of the design heat exchanger efficiency
    The value is entered in the RPN string (90 here). The #25 in the RPN string zeros this quantity if the system is not operating

44 EDI     1  1  14  41 42 20 2        -ooo ooo   3.0
  HX effectiveness diagnostic channel. Compares the average daily actual effectiveness with the average daily design
  effectiveness (constant here)

45 HDI     1  1  14  43 44 20 2        -ooo ooo   3.0  #25
  ..HX efficiency diagnostic channel. Compares the average daily actual efficiency with the average daily design
  efficiency.

STORAGE

46 TPT     1  1  10 16 0 0 0           0   100   3.0
  ..Temperature of top of preheat tank, C. Type T thermocouple on hardware channel 16.

47 TPB     1  1  10 17 0 0 0           0   100   3.0
  ..Temperature of bottom of preheat tank, C. Type T thermocouple on hardware channel 17.

48 TPA     1  1  16  0 0 0 0          0   100   3.0  #46 #47 + 2 /
  ..Average temperature of preheat tank, C. Average of top/bottom temperatures.

49 TAT     1  1  10 18 0 0 0          0   100   3.0
  ..Temperature of top of auxiliary tank, C. Type T thermocouple on hardware channel 18.

50 TAB     1  1  10 19 0 0 0          0   100   3.0
  ..Temperature of bottom of auxiliary tank, C. Type T thermocouple on hardware channel 19.

51 TAA     1  1  16  0 0 0 0          0   100   3.0  #49 #50 + 2 /
  ..Average temperature of auxiliary tank, C. Average of top/bottom temperatures.

52 TEN     1  1  10 22 0 0 0        -10    50   3.0
  ..Temperature of air in storage environment, C. Type T thermocouple on hardware channel 22.

53 TMN     1  3  10 23 0 0 0         -1    40   3.0  #53 #56 5 > *
  ..Temperature of water mains, C. Type T thermocouple on hardware channel 23.
    Note the boolean portion of the RPN string [#56 5 >] will zero the apparent mains temperature if the load
    flow rate is less than 5 l/min. This is to prevent the mains temperature from being recorded unless a draw is
    underway (since the mains pipe temp will drift to room temperature.

54 TDE     1  3  10 24 0 0 0         -1    95   3.0  #54 #56 5 > *
  ..Temperature of delivered hot water, C. Type T thermocouple on hardware channel 24.
    Note the boolean portion of the RPN string [#56 5 >] will zero the apparent delivery temperature if the load
    flow rate is less than 5 l/min. This is to prevent the delivery temperature from being recorded unless a draw is
    underway (since the delivery pipe temp will drift to room temperature.

LOAD ENERGY

55 LFTL    1  4   9 25 10 255 0        0   255   6.0  #55 1 *
  ..Total volume of fluid that passed to the load last scan in Litres. A total is to be recorded every save period.
    Assumes the analog voltage on hardware channel 25 is proportional to the number of pulses put out by the flowmeter.
    The 1 in the RPN string is the number of litres per pulse.
```

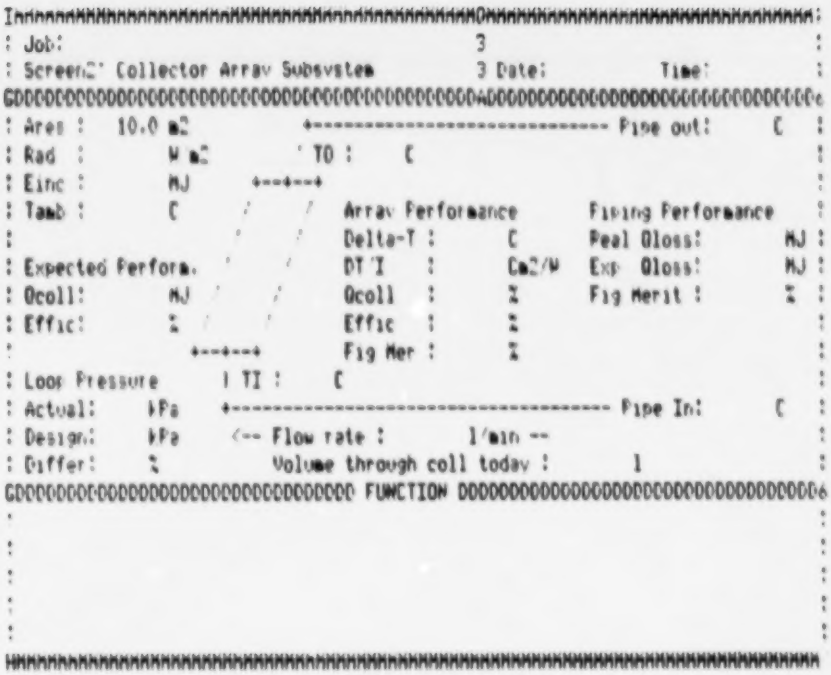
```
56 LFRT    1  3  16  0 0 0 0          0   100   3.0  #55 60 * #56.3 /
   ..Load flow rate in l/min.  Based on LFTL divided by time.  The 60 in the RPN string converts l/s to l/min.

57 LENR    0  4  16  0 0 0 0          0    20   6.1  #55 0.00419 * #54 #53 - *
   ..Total energy that has flowed into the load last scan, MJ.  This quantity totalized every save period.  Based on the water
     draw per scan (in litres) times the specific-heat/density/temperature rise of the water.  The 0.00419 in the
     RPN string is the specific-heat/density product in MJ/l.C.

58 AUXE    0  4   0 26 10 255 0        0   255   6.1  #58 0.5 *
   ..Auxiliary energy that flowed to load last scan (totaled every save period).  Channel assumes the analog voltage on hardware
     channel 26 is proportional to the number of MJ given to the auxiliary heater.  The MJ per count is the 0.5 in the RPN string.

59 SEAC    0  4  16  0 0 0 0          0    20   6.1  #57.2 #58.2 -
   ..Actual solar energy that has flowed into the load today, MJ.  Based on the difference between total energy today and the
     auxiliary energy today.

60 FSOL    1  1  16  0 0 0 0       -300   300   3.0  #59.2 #57.2 100 * /
   ..Percentage of the load energy that has been supplied by solar so far today.

61 FCN     1  4  17  25 0 0 0         0     1   3.0
$


VIRTUAL CHANNEL SCAN SEQUENCE
Note all temperatures/flow rad are measured first, then calcs performed.
$
25 1 3 4 5 6 7 30 31 32 33 34 35 21 22 23 24 46 47 49 50 52 53 54 55 56 58
2 8 9 10 11 12 13 14 15 16 17 18 19 20 26 27 28 29 36 37 38 39 40 41 42 43
44 45 48 51 57 59 60 61
-1  (-1 signifies end of sequence)

NOTES:

RESULT TYPE  (RS TYP)
   0 = REAL
   1 = INTEGER
   2 = BOOLEAN (TRUE OR FALSE RESULT)

MEASUREMENT DESTINATION VALUES (DES)
   0 = NULL
   1 = AVERAGE
   2 = NON-ZERO AVERAGE  (ZERO VALUES ARE NOT ENTERED INTO THE AVERAGE)
   3 = POSITIVE AVERAGE  (NEGATIVE OR ZERO VALUES ARE NOT ENTERED INTO AVERAGE)
   4 = TOTALIZE
   5 = TIME INTEGRAL
   6 = POSITIVE TIME INTEGRAL (TIME INTEGRAL WHEN VALUE POSITIVE ONLY)

                              FUNCTION SPECIFIC MEASUREMENT PARAMETERS
                                 1         2         3         4
VIRTUAL FUNCTION              --------- --------- --------- ---------
   1 = DC VOLTAGE              channel
   2 = AMPERES                 Not Implemented with HP3421A
   3 = TWO WIRE OHMS           channel
   4 = FOUR WIRE OHMS          low channel (high channel assumed as low channel + 10)
   5 = AC VOLTS                channel
   6 = FREQUENCY               channel
   7 = PULSE PERIOD MEASUREMENT Not Implemented with HP3421A
   8 = LEVEL SAMPLE            Not Implemented with HP3421A
   9 = TOTALIZE (D/A TECHNIQUE) channel   mV/count   max counts
```

```
  10 = TYPE T THERMOCOUPLE          channel
  11 = HP THERMOCOUPLE REF TEMP     channel (of board REF temp is desired for)
  12 = YSI 44005 THERMISTOR         Not Implemented with HP3421A
  13 = 100 OHM PLATINUM RTD         channel
  14 = DIAGNOSTIC FUNCTION          virtual   virtual    max. diff   vc field (to be used in comparison)
                                     chan1     chan2     in percent
  15 = NULL FUNCTION
  16 = PPM CALCULATOR
  17 = BOOLEAN TRANSITION           virtual
                                     chann

VIRTUAL CHANNEL FIELDS
  0 = the value of the last scan for the virtual channel
  1 = result of the destiny function selected performed from start of save period to this point in the save period
  2 = result of the destiny function selected performed from midnight up to this point in the save period
  3 = length of time (seconds) between the last scan of a channel and the present scan


SCREEN DEFINITIONS
4
SCREEN1.SCN
SCREEN2.SCN
SCREEN3.SCN
SCREEN4.SCN
SCREEN5.SCN
SCREEN6.SCN
4
```



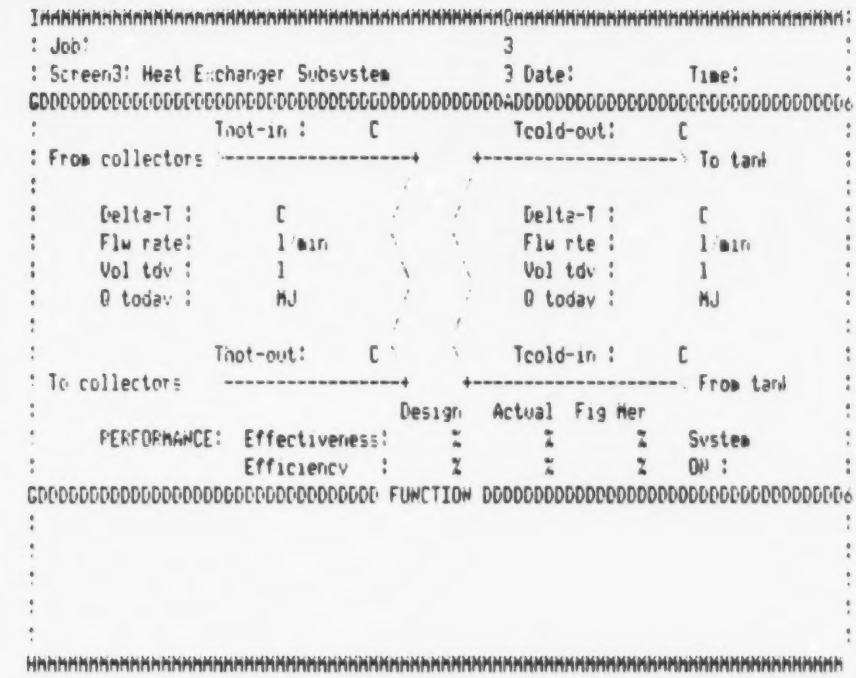
1 3 6
3 4 8
7 15 16
53 30 17
46 42 10
47 42 15
49 66 10
50 66 15
54 69 6
55.2 44 6
25 35 7

NOTE: THE BORDERS ON THE ABOVE DIAGRAM USE IBM EXTENDED ASCII CODES (8 BIT ASCII).
THE HARDCOPY REFLECTS THESE BORDERS AS 'H''s, 'D''s SINCE IT DOES NOT SUPPORT THE EXTENDED CHARACTER SET.



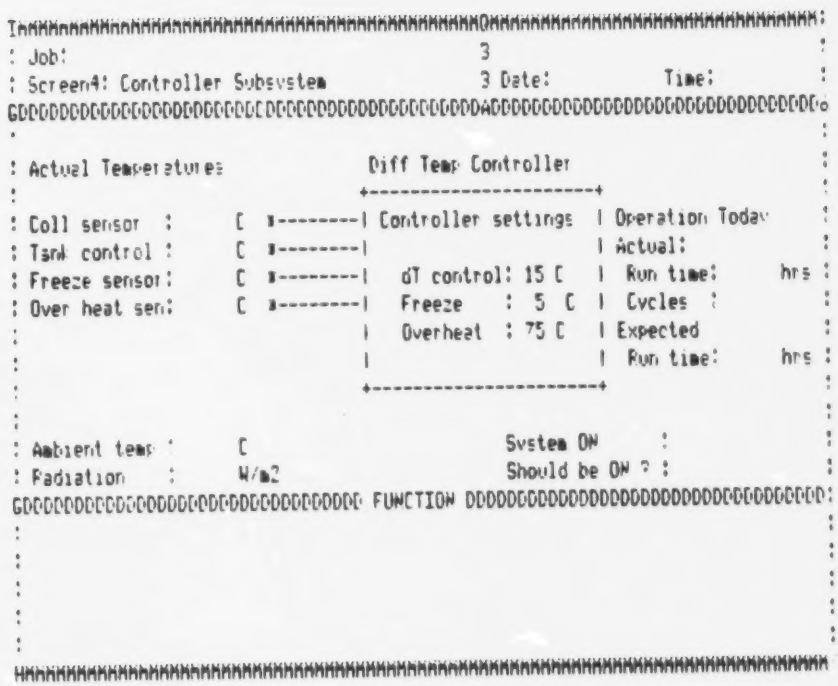
1 12 6
2.2 10 7
3 12 8
10.2 10 11
17 12 12
12 11 15
13 11 16
14 11 17
4 29 14
5 36 6
15 44 6
6 44 10
8.2 43 11
16 45 12
11 44 13
7 40 16
6.2 55 17
30 71 5
31 71 15
18.2 70 6
19.2 70 10
20 71 11

NOTE: THE BORDERS ON THE ABOVE DIAGRAM USE IBM EXTENDED ASCII CODES (8 BIT ASCII).
THE HARDCOPY REFLECTS THESE BORDERS AS 'H''s, 'D''s SINCE IT DOES NOT SUPPORT THE EXTENDED CHARACTER SET.



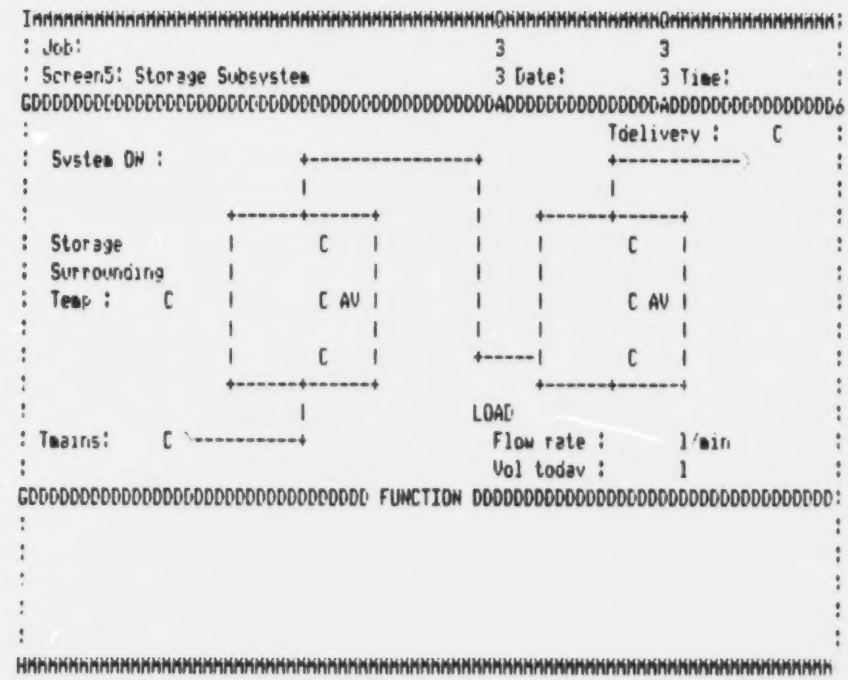
30 29 5
33 59 5
32 59 13
31 29 13
36 20 8
7 21 9
6.2 18 10
38.2 18 11
37 61 8
35 62 9
34.2 59 10
39.2 59 11
41 37 16
43 37 17
40 46 16
42 46 17
44 54 16
45 54 17
25 70 17

NOTE: THE BORDERS ON THE ABOVE DIAGRAM USE IBM EXTENDED ASCII CODES (8 BIT ASCII).
THE HARDCOPY REFLECTS THESE BORDERS AS 'M''s, 'D''s SINCE IT DOES NOT SUPPORT THE EXTENDED CHARACTER SET.



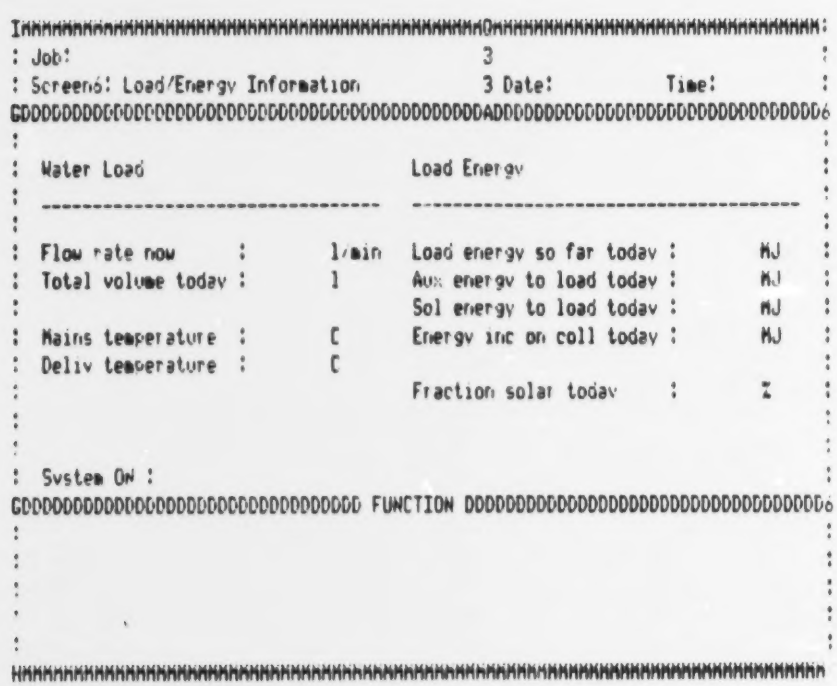
21 16 8
22 18 9
24 18 10
23 18 11
3 19 16
1 18 17
26.2 70 10
61 72 11
28.2 70 13
25 66 16
27 66 17

NOTE: THE BORDERS ON THE ABOVE DIAGRAM USE IBM EXTENDED ASCII CODES (8 BIT ASCII).
THE HARDCOPY REFLECTS THESE BORDERS AS 'H''s, 'D''s SINCE IT DOES NOT SUPPORT THE EXTENDED CHARACTER SET.



25 16 o
46 25 o
48 25 11
47 25 13
49 55 o
51 55 11
50 55 13
54 70 5
52 11 11
53 11 16
56 61 16
55.2 58 17

NOTE: THE BORDERS ON THE ABOVE DIAGRAM USE IBM EXTENDED ASCII CODES (8 BIT ASCII).
THE HARDCOPY REFLECTS THESE BORDERS AS 'M''s, 'D''s SINCE IT DOES NOT SUPPORT THE EXTENDED CHARACTER SET.


: Job: 3 :
: Screen6: Load/Energy Information 3 Date: Time: :
Water Load
Load Energy
Flow rate now : l/min
Load energy so far today : MJ
Total volume today : l
Aux energy to load today : MJ
Sol energy to load today : MJ
Mains temperature : C
Energy inc on coll today : MJ
Deliv temperature : C
Fraction solar today : %
System ON :
FUNCTION

56 28 9
55.2 27 10
53 28 12
54 28 13
57.2 67 9
58.2 67 10
59.2 67 11
2.2 67 12
60 70 14
25 16 17

NOTE: THE BORDERS ON THE ABOVE DIAGRAM USE IBM EXTENDED ASCII CODES (8 BIT ASCII).
THE HARDCOPY REFLECTS THESE BORDERS AS 'H''s, 'D''s SINCE IT DOES NOT SUPPORT THE EXTENDED CHARACTER SET.

CHAPTER 6

SENSOR CONNECTIONS

This section contains a brief description of the proper sensor connection methods for the SCP.

The HP 3421A "Operating, Programming, And Configuration Manual" contains a more elaborate discussion of connections to the HP 3421A. This section is intended only as a brief reference. All user's of the SCP package should read the HP 3421A manual before connecting sensors to the device.

6.1 The HP 3421A Channel Arrangement

The HP 3421A is equipped with 30 input channels in the basic SCP package. Three HP option 44462A 10 channel multiplexer cards are installed in the HP mainframe.

All multiplexer channels are configured as general purpose inputs except channels 20 and 21, which are configured as actuator outputs. As a result these two channels can NOT be used as input channels. The HP 3421A will issue an error message if an attempt is made to use these channels as inputs.

Each board is equipped with a strain relief assembly, to reduce the possibility of cable breakage or damage to the HP 3421A. The strain relief assemblies are removed to obtain access to the screw terminal blocks for each channel.

When the strain relief assemblies are removed, the screw terminal blocks become accessible. Each block has two knurled machine screws that fasten the terminal block to the

multiplexer board. These knurled screws are undone to allow the terminal block to be removed from the 3421A assembly. Sensor wire connection is made easier if the terminal blocks are taken off of the multiplexer boards.

Channel Numbering

The 30 channels are segmented into three groups of ten channels, one group per multiplexer board. Channels on each board are numbered 0..9, and the three boards are numbered 0 to 2. Two of the ten channel boards are located on the left of the back panel, and the third is located on the right. The board numbers are written in large numerals to the side of each board.

The channel numbers range from 0 to 29. The 10 channels on board 0 are numbers 0..9, the 10 channels on board 1 are numbers 10..19, and the 10 channels on board 2 are numbers 20..29.

Channel Polarity

Each input channel has a positive or high terminal (marked "H"), and a negative or low terminal (marked "L"). For example, 0-H would correspond to the positive side of input channel 0, while 0-L would correspond to the negative side of input channel 0.

Common Rails

Each of the three terminal block assemblies have screw terminals for the 10 channels on the board, as well as one pair of terminals that are the common rails. This pair does not have a number marked on the terminal block as with the 10 regular input channel terminal pairs.

This common pair is marked "L" and "H" (with no number), and is the analog bus common terminals. These terminals MUST not be connected to anything for proper operation of the SCP package.

6.2 Thermocouples

Type-T thermocouples are a junction of a copper wire and a constantan wire. The thermocouple junction itself must be either welded or soldered. A mechanically and electrically solid junction is essential for proper operation of the sensor.

The thermocouple junction should be about 6 mm in length, and tightly twisted together in a rat-tail splice before soldering. Only electrical (resin core) solder should be used on the junction. Standard plumbing type acid core solder must not be used. Ensure enough heat is applied to the junction when soldering to obtain a smooth clean layer of solder. Constantan is more difficult to solder than copper.

The negative or LOW side of a channel is connected to the CONSTANTAN wire (the white one), and the positive or HIGH side of the channel is connected to the COPPER wire. If using small gauge wire it may be wise to double or quadruple the bare wire before inserting it under the HP 3421A screw terminal.

If the thermocouple polarity is reversed the temperature reading will increase as the real temperature drops. To ensure the thermocouple is attached correctly, heat the thermocouple up and ensure the measured result also increases.

When attacning the junction to a copper pipe or other surface, ensure the thermocouple is in good mechanical contact with the surface. A clamp of some sort should be used where possible. Insulation should also be applied to the exterior of the surface/thermocouple to ensure the reading obtained is as close to the surface temperature as possible and not affected by the surrounding air temperature.

It is always wise to manually verify that a given temperature reading corresponds to the desired thermocouple. This can be done at set-up time using a thermos ice-bath, and manually inserting each thermocouple into the ice bath, one at a time. With two persons and a pair of walkie-talkies, each sensor can placed in the bath and the reading on the SCP screen viewed. This test ensures that the thermocouple is accurate (ice-bath should yield +/- 0.1 C), and that each sensor is in tne correct spot.

6.3 Pyranometers

A solar radiation pyranometer is required to measure the solar energy incident on the array.

Pyranometers normally have a two wire output, one wire positive and the other negative. The positive wire must be connected to the HP channel HIGH terminal, while the negative terminal is connected to the HP channel LOW terminal.

A pyranometer output is usually very small, in the order of 80000 W/m2 per volt. The calibration constant for the pyranometer being used must be entered in the CONFIG.MON file to ensure meaningful results.

The pyranometer must be mounted in the plane of the collector array. This line up is fairly critical since all efficiency measurements will be based on this one number. Ensure the pyranometer is carefully lined up, not shaded by anything, and is securely mounted.

6.4 Flowmeters

The SCP default CONFIG.MON file assumes integrating type flowmeters will be used. These flowmeters yield a pulse every unit of fluid that passes through them. One pulse per litre or per ten litres are common outputs.

If virtual function number 9 is used to totalize flowmeter pulses (as in the default files), the SCP package assumes the DC voltage on the flowmeter channel varies in proportion to the number of pulses that have occurred, and that the voltage rolls-back to 0.0 volts after a certain number of pulses. This technique was installed to allow an 8 bit digital counter to directly count the incoming pulses, and have this counter drive a digital-to-analog converter circuit which in turn supplies the DC output voltage. When the count reaches 255 (the maximum), the output voltage would be (say) 2.55 volts. When the count is 0, the output is 0.0 volts. The voltage inside this range varies linearly at 0.01 volts per pulse. This circuit, or alternate have yet to be acquired before using these flowmeters with the SCP.

The positive voltage output of this D/A circuit would be

connected to the HIGH HP terminal, and the negative to the LOW HP terminal.

The litres of fluid per pulse, the number of counts full scale, and the voltage change per pulse are all inputs to virtual function 9 (see Section 4).

Frequency output flowmeters may also be accommodated by using the "FRQ" (frequency) measurement function in the SCP software.

6.5 Pressure Sensors

The default solar system CONFIG file is set up to accommodate a voltage output pressure sensor.

If the kPa per volt of the sensor is other than 200, the CONFIG.MON file will require modification.

The positive lead of the sensor is to be connected to the HIGH terminal of the HP channel, and the negative lead to the LOW terminal of the channel.

6.6 Event Sensing

Event sensing or device status may be measured in many different ways. A common method is to provide a DC voltage that represents the state of the device.

With a DC power source (power supply or battery), and a relay connected across the power input of a pump (or other device), a DC output voltage may be fed to the SCP package.

Since the RPN calculator function (see Section 4) will allow any boolean condition to be evaluated, any tolerance or ON/OFF threshold may be entered. For example, the DC voltage may be 9.0 VDC when the pump is ON, and 0.0 volts when the pump is off. In this case a reasonable ON/OFF threshold would be 4.5 volts.

The base SCP CONFIG file assumes a DC voltage will be supplied to indicate the ON/OFF status of the solar system. If the voltage exceeds 4 volts, the system is considered to be ON, otherwise it is considered to be OFF.

SOLAR COMMISSIONING PACKAGE OPERATIONS MANUAL
APPENDIX A : ERROR MESSAGES

This section describes the various error messages produced by the MONITOR program.

An error condition is not to be confused with a diagnostic condition. An error condition occurs if a malfunction of the SCP measurement package itself is discovered. Errors are not generated as a result of the performance of the system being measured.

The major types of error condition are segmented as follows:

1. Computer system errors.
2. Measurement hardware errors.
3. Measurement software errors.
4. RPN calculator errors.
5. Internal errors.

An error message is always issued by the SCP. The message issued is at most 20 characters in length. The following section elaborates on these simple messages, and where practical, refers the reader to an appropriate reference or discusses possible solutions to the problem.

Each error condition has a number associated with it. This number is unique for the message, and is also printed in the error message file on disk. The error messages that follow are arranged with the error numbers in increasing order.

-Power Failure Error

5000 : message = "Pwr failure detected"
-An SCP computer system power failure was detected. The SCP will log this error condition every time a the program initializes and also finds the SYSTEM.LOG file already in existance on disk. Only if SYSTEM.LOG has not been created does the system assume no power failure occurred.

Computer File System Errors

Error numbers 5103 to 5115 indicate file system errors. The name of the file being processed can be determined through the error number presented in the error message. The error parameter number is the integer which follows the "vc:" number in the error message.

The error parameter will be one of the following:

1000; In CONFIG.MON file.
1004; During read of ERROR.MES.
1005; During read of DIAGS.MES.
1006; During save of averages/totals.
1007; During write to ERROR.MES.
1008; During write to DIAGS.MES.
1010; In procedure: append-a-line-to-a-file.
1011; In DOS temporary file chr(0).
1012; During proc to transfer infile-outfile.
1014; In hourly file-during calcs of avg.
1015; During proc fill-temp-array-from-disk.
1020; In SYSTEM.LOG file.

5103 : message = "Invalid file oper"
-An illegal disk file operation was detected.

5107 : message = "Invalid file name"
-DOS detected an illegal file name. Perhaps the default disk drive specified in the CONFIG.MON file is illegal?

5108 : message = "Disk or dir full"
-The diskette directory is full (too many entries), or the diskette itself is full. Either replace the diskette or perform a "CHKDSK".

5110 : message = "File not found"
-DOS could not find the file.

5113 : message = "I/O to unopen file"
-Attempt to read or write to a file that is not opened.

5114 : message = "Illeg data format"
-The data is in an illegal format for the read statement. Could be illegal characters in a number (e.g.,10a is an illegal number).

5115 : message = "File line too long"
-A line in a file was too long to be properly processed.

5200 : message = "DAILY.SUM not found"
-The DAILY.SUM file could not found.

Measurement Hardware Errors

6050 : message = "IEEE card failed"
-This is a general message indicating the IEEE card failed. If this message is issued record the exact error number in the message. The error number will not be 6050, but rather a number between 6000 and 6039 that indicates the exact point of failure of the IEEE function. The IEEE interface will fail periodically due to unreliability of the software driving the interface. Unless the failures become excessive, disregard them. MONITOR disregards all data obtained under error conditions.

6040 : message = "Char ovflw IEEE read"
-The HP 3421A sent more than 25 characters as a measurement result.

6041 : message = "Tme out in IEEE read"
-The HP 3421A took too long to respond.

6100 : message = "HP abnorm condition"
-This indicates the HP 3421A detected an abnormal condition. When the SCP software detects this error, the HP 3421A is polled to determine the exact nature of the error. As a result another error number/message should follow this one detailing the problem.

6102 : message = "HP issued SRQ"
-The HP 3421A issued a service request message. The SCP software handles this condition.

6103 : message = "HP pwr on reset det"
-The HP 3421A indicated a power on reset was detected [in the HP not the IBM]. This message occurs periodically due to unsolved problems with the IEEE interface. Disregard unless excessive.

6104 : message = "Measurement time out"
-The HP 3421A did not respond with the result of a measurement in time. Three seconds are allowed for every measurement.

The following errors indicate the HP 3421A has failed in some form. Consult the HP reference manual for more information.

6110 : message = "Low HP3421 battery"
-The HP 3421A has indicated that its battery is low. Could be caused from leaving the power switch on while unplugged for long periods. Plug the HP back into HYDRO and allow to recharge.

6111 : message = "Illeg fun w/low batt"
-An attempt to execute a function that cannot be undertaken with a low battery.

6112 : message = "Cal ram checksum"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6113 : message = "HP rom 1 checksum"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6114 : message = "HP rom 0 checksum"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6115 : message = "HP A/D slope error"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6116 : message = "HP uP rom failed"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6117 : message = "HP rom U504 failed"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6118 : message = "HP rom U503 failed"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6119 : message = "HP 10 Mohm test fail"
-the resistance of the HP3421A common analog rails is/was less than 10 M ohms. Check that nothing is connected to the common bus and ensure all voltages connected to the HP are within range.

6120 : message = "HP self test fail"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6150 : message = "HP Evnt occ MN,ML,MH"
-should not occur in the SCP.

6151 : message = "HP trig Fo assert"
-should not occur in the SCP.

6152 : message = "HP talk with no data"
-The HP was asked to transmit data when no data was available.

6153 : message = "HP Inval cmd syntax"
-The syntax of the command sent to the HP was undefined.

6154 : message = "HP No option in slot"
-should not occur in the SCP.

6155 : message = "HP T3 with no list"
-should not occur in the SCP.

6156 : message = "HP load list > 30 ch"
-should not occur in the SCP.

6180 : message = "HP Inval cal fun/rng"
-should not occur in the SCP.

6181 : message = "HP Inval cal number"
-should not occur in the SCP.

6182 : message = "HP inval cal signal"
-should not occur in the SCP.

6183 : message = "HP Inval cal zero"
-should not occur in the SCP.

6184 : message = "HP cal atmpt no ram"
-should not occur in the SCP.

6185 : message = "HP bad cal ram"
-should not occur in the SCP.

6186 : message = "HP A/D error"
-HP 3421A internal hardware failure.
Refer to HP service manual.

6200 : message = "HP +8.888888e08"
-The HP sends this number when it detected an error during a measurement. The SCP should not report this since it polls the HP to determine the exact nature of the error and replaces this error message with the real one. Report it as a bug if detected.

6201 : message = "HP signal overrange"
-Overrange on a channel. Could be a thermocouple open circuit, mismatch of sensors, etc. Check all sensor connections and voltages.

6202 : message = "HP sent illegal str"
-The HP 3421A sent a string that was undecodable as a number. This could occur infrequently due to noise. If frequent report it as a bug.

6203 : message = "HP chan<0 or >29"
-An illegal channel has been specified. Check CONFIG.MON for good channel numbers (between 0 and 29).

6204 : message = "HP illegal range"
-should not occur in the SCP.

6205 : message = "HP AZ para illegal"
-Illegal auto zero parameter sent to the HP.
Only Z0 or Z1 strings can be sent.
Check CONFIG.MON file, HP set up line.

6206 : message = "HP Ndigits illegal"
-Illegal resolution parameter sent to the HP.
Only N3, N4 or N5 strings can be sent.
Check CONFIG.MON file, HP set up line.

6208 : message = "Reftmpvar <10 or >40"
-should not occur. If so report a bug.

6209 : message = "Actual Tref <10or>40"
-The HP reference temperature was out of
the range given above. This is a message
issued by the HP when it automatically
compensates for the thermocouples. If the room
temperature is out of the range given above
the SCP should not be operating. If not,
the HP hardware may have failed.

6250 : message = "HP cant measure this"
-Amps, Pulse period measurement, level sample,
or other illegal function was specified. Check
CONFIG.MON contents for valid function numbers.

6251 : message = "HP illegal function"
-HP did not recognize the function requested.

6252 : message = "YSI R<55 or >56K"
-YSI44005 resistance out of the legal
range for -30 < T < 150 C. MONITOR version
1.5 should never issue this message.

6253 : message = "Denom of ln pol = 0"
-In YSI44005 routine the curve fit denominator
was out of range. MONITOR version
1.5 should never issue this message.

6254 : message = "RTD res out of range"
-RTD resistance was not between 85 ohms and 160 ohms.
Bad sensor, connections or extreme temperatures.

6260 : message = "D/A count illegal"
-The D/A voltage for totalize function 9 indicated
the digital counter had a count either less than
zero or greater than the maximum full scale count.

6261 : message = "Illegal vc field"
-A VC field other than 0,1 or 2 was specified. Check all RPN formulae and display screen VC number list.

Measurement Software Errors

7001 : message = "Illeg vc# in scanseq"
-An undefined VC number was found in the scan sequence. Check the CONFIG.MON file.

7002 : message = "Trans on non-boolean"
-Request to determine transitions on a non boolean channel? Check the CONFIG.MON file.

7010 : message = "Vc val < abs minimum"
-VC value lower than the absolute minimum expected. Could be bad sensor or bad data in CONFIG.MON file. Ensure lower limit specified in file is OK.

7020 : message = "Vc val > abs maximum"
-VC value higher than the absolute maximunm expected. Could be bad sensor or bad data in CONFIG.MON file. Ensure upper limit specified in file is OK.

7030 : message = "Vc# as transpar illg"
-The VC number specified as a parameter in another VC for the transition function was illegal.

7040 : message = "Measfun <0 or >17"
-Illegal virtual function encountered. Bad data in CONFIG.MON file.

7041 : message = "Scrn X-coord illegal"
-Screen coordinate out of legal range (1..80). Check screen definitions and re edit.

7042 : message = "Scrn Y-coord illegal"
-Screen coordinate out of legal range (1..25). Check screen definitions and re edit.

7050 : message = "VC for field nonexis"
-The routine that recalls the value of a VC field attempted to recall an undefined element.

RPN Calculator Errors

7100 : message = "RPN undecodable vc#"
-RPN detected a VC number that was undecodable as a number. Bad characters ? Check CONFIG.MON file.

7102 : message = "RPN Insuff operators"
-Not enough operators were specified in a formula to exaust the data. Bad formula.
Find VC number from error message and edit CONFIG.MON.

7103 : message = "RPN stack ovrflw[20]"
-Attempt to push too many items on the stack.
Find VC number from error message and edit CONFIG.MON.

7104 : message = "RPN 0 char in form"
-The RPN calculation function was specified but no characters exist in the formula?
Find VC number from error message and edit CONFIG.MON.

7120 : message = "RPN vc value bad"
-Not really an error. This should not be logged in the ERROR.MES file. Indicates that one of the VC numbers needed for a computation had an undefined value. Should result only in the calculation result being undefined.

7121 : message = "RPN illeg numer con"
-An undecodable numerical constant was found in an RPN string. Likely bad characters in a string.
Find VC number from error message and edit CONFIG.MON.

7125 : message = "RPN unary op < 1 #"
-A unary operator was found with no items on the stack ? Find VC number from error message and edit CONFIG.MON.

7126 : message = "RPN binary op < 2 #"
-A binary operator was found with less than two items on the stack ?
Find VC number from error message and edit CONFIG.MON.

7130 : message = "RPN divide by zero"
-RPN was asked to divide by zero.
Find VC number from error message and edit CONFIG.MON.

7131 : message = "RPN nonpos to power"
-RPN was asked to raise either zero or a negative number to a power.
Find VC number from error message and edit CONFIG.MON.

7132 : message = "RPN root of negative"
-RPN was asked to take the square root of a negative number. Find VC number from error message and edit CONFIG.MON.

7133 : message = "RPN ln of negative"
-RPN was asked to take the natural log of a number.
Find VC number from error message and edit CONFIG.MON.

7134 : message = "RPN log of negative"
-RPN was asked to take the log (base 10) of a number.
Find VC number from error message and edit CONFIG.MON.

7135 : message = "RPN ATN when cos=0"
-Attempt to compute arc tangent when cosine of the angle is zero.
Find VC number from error message and edit CONFIG.MON.

7136 : message = "RPN exp of # > 70"
-Attempt to raise e to a power greater than 70.
Find VC number from error message and edit CONFIG.MON.

General Messages

7200 : message = "Save period illegal"
-An illegal save period was specified. Check CONFIG.MON contents and enter a new value.

7201 : message = "Scnseq lod nexst vc"
-An attempt to load the scan sequence array with a VC that does not exist. Edit the scan sequence in the CONFIG.MON file.

Internal Errors

8100 : message = "RPN op < 1 or > 23"
-RPN thinks it found an operator that isnt even defined. This should be reported as a bug.

8110 : message = "VC destiny illegal"
-An illegal VC destiny was discovered by the program.

The backspace key deletes the character to the left of the cursor. The DELETE key deletes the character under the cursor.

Control-X will delete the line the cursor is on. Control-Z will delete the text on the cursor line between the cursor point and the end of the line. Control-home will position the cursor to the top of the file. Control-end will position the cursor to the bottom of the file. Function key "F5" advances the cursor to the beginning of the next line, and "F6" advances the cursor to the end of the current line.

Command Mode

To enter COMMAND mode from VISUAL mode press the ESC key. To return to visual mode from COMMAND mode press "V<return>".

In command mode, pressing "EX<return>" will EXIT back to DOS and save the new copy of the edited file under the original filename. A backup copy of the original file will automatically be stored in a file with the same name and an extension of ".BAK".

If "EQ<return>" is pressed in command mode, VEDIT will return to DOS without saving any of the changes back to disk.

SOLAR COMMISSIONING PACKAGE OPERATIONS MANUAL
APPENDIX B : USING VEDIT - THE FULL SCREEN EDITOR

This Appendix contains a very brief introduction to the full screen editor, VEDIT, by Compuview Products Inc., Ann Arbor, MI, USA.

The VEDIT user manual, some 200 pages, should be reviewed if this short summary is insufficient for the operator.

The editor is contained in a file called "ED.COM" on the SCP disks. To edit a file, ensure the drive A disk contains the ED.COM file, and set the default DOS drive as A:. Put the disk containing the file to be altered in drive B:. Type the following command;

ED B:yourfile.nam <return>

Where yourfile.nam is the DOS filename to be edited. Note that to edit a file on another drive, change the B: that preceeds the filename above.

After a few seconds, VEDIT will scroll the screen up and display the first page of the file.

There are two modes in running the VEDIT program. The first is VISUAL mode. In this mode text can be inserted, the cursor can move about on the screen, and so on. The second mode is the COMMAND mode. This allows advanced commands, as well as EXIT or QUIT command execution.

Visual Mode

Wnen VEDIT starts you are in VISUAL mode. The bottom row of the screen in VISUAL mode contains the VEDIT status line. Among other things, this shows the current cursor Y,X coordinates (Useful for determining the coordinates of the VC fields).

The four IBM cursor control keys (the arrows) will move the cursor around on the screen.

Pushing the IBM "Insert" key will toggle the insert mode. If you were in insert mode, pressing INSERT will cancel the mode. If not in insert mode, pressing insert will enable it. If a printable character is pressed when in insert mode, the character will be inserted into the text. If not in insert mode, pressing a character will result in an overstrike.

SOLAR COMMISSIONING PACKAGE OPERATIONS MANUAL
APPENDIX C : USING MICRO WATSUN

This Appendix introduces the IBM PC version of the WATSUN computer program.

The program, termed "micro-WATSUN", was purchased from the WATSUN User Service as part of the SCP development, and has been installed on the IBM PC. The present version of the program contains only the DHWA module of WATSUN.

Since the program has not yet been fully tested, it is NOT supported by the WATSUN User Service. During program installation, several program bugs were found and have been corrected. The program does run, but Sheaff And Associates Engineering Inc. take no responsibility for the accuracy or reliability of the software.

Diskettes

Two diskettes are provided containing the software. The first, titled "WATSUN IBM PC SOURCE CODE", contains the BASIC program source code, object modules, and several data files. The second disk, titled "WATSUN RUN DISK" contains the executable file "WATSUN.EXE", system data files, and weather data for Montreal, 1970.

Manuals

The WATSUN User Service have provided two manuals. The first covers use of the micro WATSUN program and the second covers the operation of the program set. These manuals have been supplied under separate cover. No attempt will be made here to duplicate the official WATSUN documentation.

Running WATSUN

To run the WATSUN program, boot the IBM PC, insert the WATSUN RUN DISK in drive A:, and type "WATSUN". After several seconds a menu will be displayed and will allow the user to run all aspects of the program (e.g., interactive, simulation, etc.).

Creating SCP Weather Data Files

A program has been written to allow the measured radiation and ambient temperature data from the SCP to be converted into the format required by the WATSUN program. The program, called "WATWEATH.EXE" is written in IBM Pascal.

The required format is outlined in the documentation provided by the WATSUN User Service. WATSUN requires weather files containing hourly records of incident solar radiation in kJ/hr.m2, ambient temperature in C, and the cosine of the incident angle for the hour.

The WATWEATH program will create a WATSUN "titled" weather data file containing the necessary parameters. The radiation is read from the SCP output data files, as is the ambient temperature. The cosine of the incident angle is computed based on the models used in the official mainframe version of WATSUN.

To run the WATWEATH program, insert the WATSUN run disk in drive A:, and type "WATWEATH<return>".

The WATWEATH program will ask the operator questions about the transfer. The required information is:

1. First date of the transfer (e.g.,Ø2JUN84).
2. Last date of the transfer (e.g.,Ø3JUN84).
3. DOS filename for output file.
4. Disk to search for input files.
5. Nearest city.
6. Province.
7. Latitude of city.
8. Collector slope.
9. Collector azimuth.
10. Fields of the stored SCP data that correspond to the solar radiation and the ambient temperature, as well as a radiation conversion factor to obtain kJ/hr.m2.

The program will then search for all files between the two dates, read the data contained in the files, and write the output file ready for WATSUN to read.

As the program runs, the date being processed will be printed. The hours found will also be printed. If an hour is printed followed by a "T", it means the ambient temperature field was

found for that hour (i.e., a good value). Likewise for radiation if an "R" is printed.

If the program cannot find data for a particular hour of the day, the value for the same hour yesterday is assumed. If the program cannot find the first days data, or hours within the first day, the ambient temperature is taken as 5.0 C and the radiation is assumed as 0.0 kJ/hr.m2.

Note that this program produces TILTED weather data files since the SCP radiation data is already adjusted to be that which fell in the plane of the collectors. The filename chosen for the output file must have an extension of ".TLT" in line with the WATSUN convention.

When the transfer is complete, WATSUN can be run. Make sure the simulation block of the WATSUN input data file contains the correct location, year, and most importantly the start and ending day of the simulation. If these two days are not correct WATSUN may not run.

A listing of the WATWEATH source code has been included in the SCP program reference manual. On disk, the source file name is WATWEATH.PAS.